



TECNOLÓGICO
NACIONAL DE MÉXICO

TESSFP
TECNOLÓGICO DE ESTUDIOS SUPERIORES
SAN FELIPE DEL PROGRESO

TECNOLÓGICO DE ESTUDIOS SUPERIORES DE SAN FELIPE DEL PROGRESO

CÓMPUTO EN LA NUBE

PRACTICAS

DOCENTE:

LIZETH JOSE ANGELES

ALUMNO:

DAVID FELIPE ROMAN

INGENIERÍA INFORMÁTICA

NOVENO SEMESTRE

INDICE

INTRODUCCIÓN	4
Requerimientos para realizar la práctica	4
CREAR CUENTA/INICIAR SESIÓN	5
HABILITAR AUTENTICACIÓN.....	10
HOSTING	13
HABILITAR BASES DE DATOS	24
VALIDACIONES	27
ESTILOS.....	30
Análisis	33
Ilustración 1 Búsqueda de firebase	5
Ilustración 2 Pagina inicial.....	5
Ilustración 3 Sesion Iniciada	6
Ilustración 4 Asignar nombre al proyecto	6
Ilustración 5 Terminos y condiciones	7
Ilustración 6 Crear Proyecto.....	7
Ilustración 7 Habilitar google analitycs	8
Ilustración 8 Seleccionar País	8
Ilustración 9 Creando Proyecto.....	9
Ilustración 10 Pagina principal del proyecto	10
Ilustración 11 Autenticación	10
Ilustración 12 Correo y contraseña	11
Ilustración 13 Autenticacion Habilitada	11
Ilustración 14 Auteticacion finalizada	12
Ilustración 15 Configuracion del proyecto	13
Ilustración 16 Datos del proyecto	13
Ilustración 17 Apps.....	14
Ilustración 18 Registro de aplicación	14
Ilustración 19 Ejecucion del cmd.....	15
Ilustración 20 comando npm install firebase	15
Ilustración 21 comando para instalar las herramientas de firebase.....	16
Ilustración 22 Login Exitoso.....	16
Ilustración 23 Inicio de sesión con correo}	16
Ilustración 24 inicio de sesion exitoso.....	17
Ilustración 25 Carpeta creada	17
Ilustración 26 Codigo HTML	18
Ilustración 27 código app.js.....	19
Ilustración 28 Inicializamos firebase	20

Ilustración 29 Configuracion Inicial	21
Ilustración 30 Carpeta Public	21
Ilustración 31 Carpetas 404 e index.....	21
Ilustración 32 subir proyecto con deploy.....	22
Ilustración 33 inicio de sesión	22
Ilustración 34 Usuarios Guardados	23
Ilustración 35 crear base de datos	24
Ilustración 36 Reglas de Seguridad	25
Ilustración 37 Crear colección	25
Ilustración 38 Datos guardados en la base de datos.....	26
Ilustración 39 Validacion campos vacíos.....	27
Ilustración 40 código de validación	27
Ilustración 41 Validacion de día	28
Ilustración 42 Código de validación	28
Ilustración 43 Validacion de reservación	28
Ilustración 44 Codigo de validacion	29
Ilustración 45 Inicio de sesion con estilos	30
Ilustración 46 Pagina de reservaciones con estilos.....	31
Ilustración 47 Formato de Fecha.....	32
Ilustración 48 Formato de Hora	32

INTRODUCCIÓN

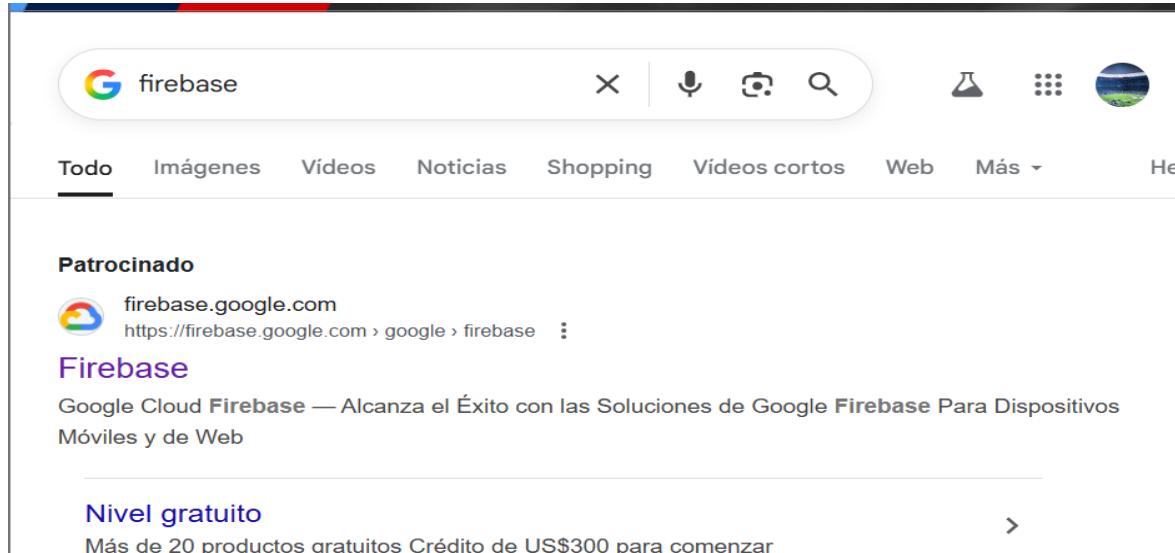
Este manual tiene como objetivo explicar el desarrollo y funcionamiento de un sistema de reservación de laboratorios utilizando tecnologías web y servicios en la nube. A lo largo del documento se describen los archivos principales del proyecto, su estructura y la lógica implementada para permitir el registro de usuarios, el inicio de sesión, la realización de reservas y la visualización de las mismas. Se utilizó Firebase como backend para manejar la autenticación de usuarios y el almacenamiento de datos en Firestore. Además, se aplicaron buenas prácticas de validación y diseño para asegurar una experiencia sencilla y funcional para el usuario final. Este sistema está pensado para ser práctico, accesible y fácil de mantener.

Requerimientos para realizar la práctica

- ✓ Contar con una computadora con conexión a internet.
- ✓ Tener instalado un editor de código (por ejemplo, Visual Studio Code).
- ✓ Disponer de una cuenta activa en Firebase.
- ✓ Crear un proyecto en Firebase y habilitar los servicios de Authentication (correo y contraseña) y Firestore.
- ✓ Tener instalados Node.js y npm (gestor de paquetes).
- ✓ Instalar la herramienta de Firebase CLI utilizando el comando: `npm install -g firebase-tools`.
- ✓ Iniciar sesión en Firebase desde la terminal con el comando: `firebase login`.
- ✓ Inicializar el proyecto con `firebase init`, seleccionando los servicios necesarios.
- ✓ Asegurarse de contar con los archivos del proyecto (HTML, CSS, y JavaScript) correctamente estructurados.
- ✓ Utilizar el comando `firebase deploy` para publicar el sistema en la nube.

CREAR CUENTA/INICIAR SESIÓN

Paso 1 : Buscamos en el navegador “FIREBASE” e ingresamos al primer link.



Paso 2: veremos la página principal de firebase nos ubicaremos hasta la parte posterior de la página y veremos el siguiente contenido, seleccionaremos la opción de comenzar.

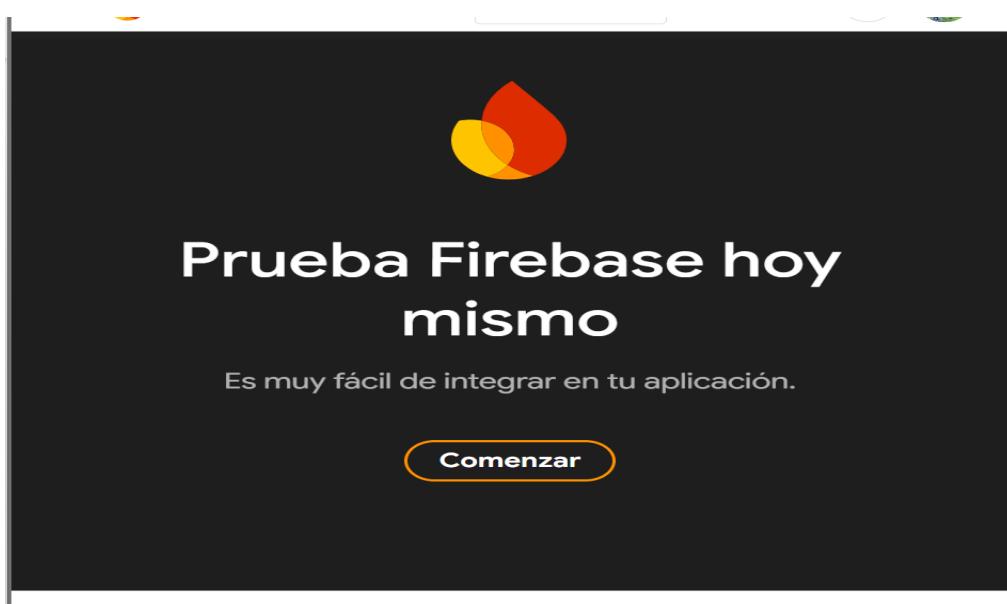


Ilustración 2 Pagina inicial

Paso 3 : Si somos usuarios nuevos nos pedirá iniciar sesión en mi caso ya tenía una cuenta asignada así que nos aparecerá la siguiente página



Ilustración 3 Sesión Iniciada

Paso 4 : Desde esta página podremos crear un nuevo proyecto > asignaremos un nombre al proyecto

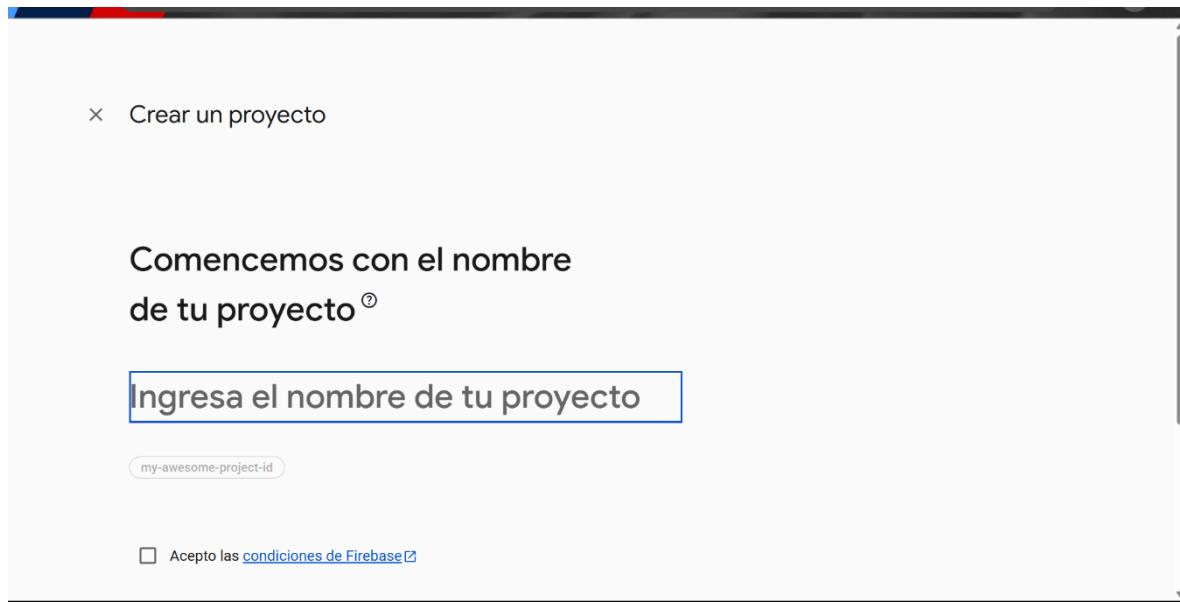


Ilustración 4 Asignar nombre al proyecto

Aceptaremos términos y condiciones y procedemos a continuar.

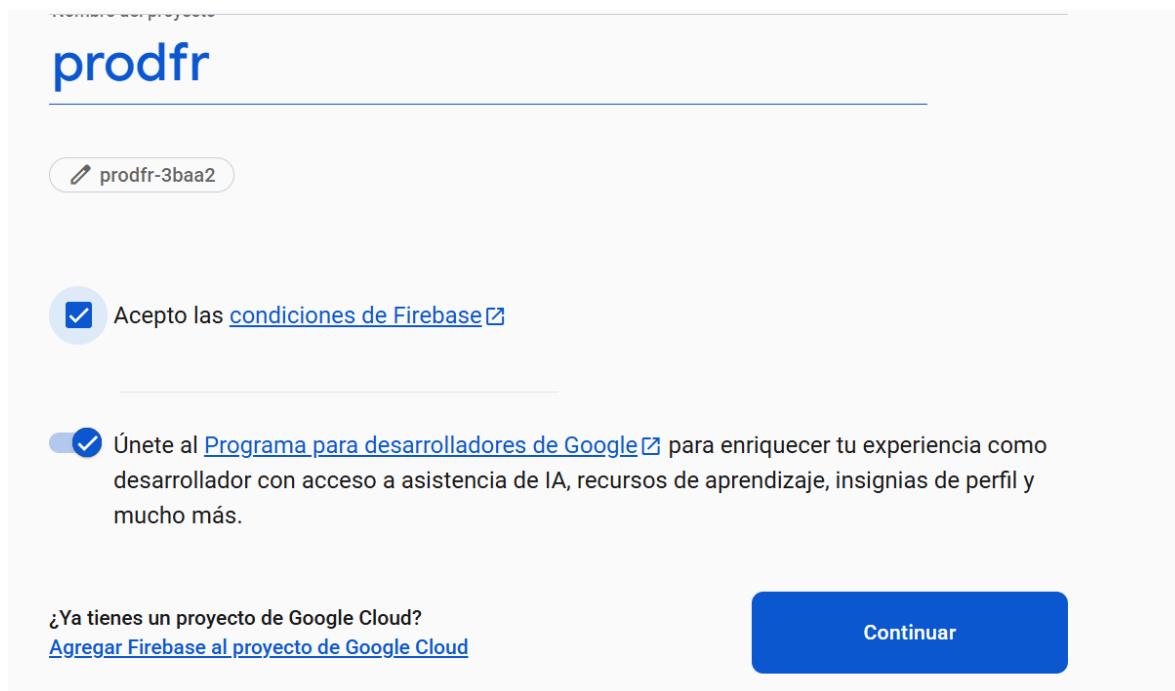


Ilustración 5 Términos y condiciones

Continuando con el proceso de creación de proyecto dejamos por default esta pagina y seguimos continuando.

Crear un proyecto

Estadísticas y recomendaciones de campañas para mantener la participación de los usuarios con Firebase Cloud Messaging

Generación de esquemas y exploración de datos en lenguaje natural con Firebase Data Connect

Habilitar Gemini en Firebase
Recomendado

Renuncias de responsabilidad:

Gemini en Firebase aún está en desarrollo y puede proporcionar información imprecisa. Prueba el código que genera antes de usarlo en tus apps. Gemini puede interactuar con tus datos de Firebase y adaptar las respuestas a tu app, y puede usar tus instrucciones o datos para el entrenamiento. [Obtén más información sobre cómo entramos nuestros modelos](#) El uso de Gemini en Firebase está sujeto a las [Condiciones del Servicio de Google](#) y a la [Política de Uso Prohibido de IA Generativas](#)

[Anterior](#)

[Continuar](#)

Ilustración 6 Crear Proyecto

Seguimos continuando hasta llegar al siguiente apartado donde deberemos de habilitar Google analytics, esto sirve para medir, analizar y optimizar el comportamiento de los usuarios en tu app.

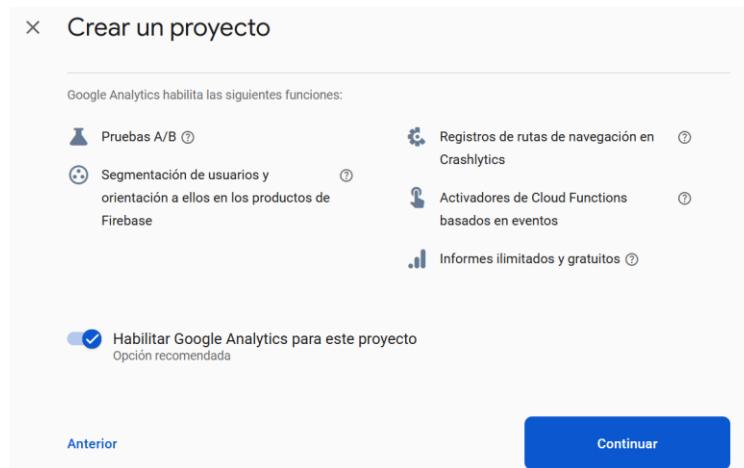


Ilustración 7 Habilitar google analytics

Al dar en continuar nos mandara a la siguiente pantalla donde deberemos seleccionar el país de analytics al igual que aceptar lo términos y condiciones

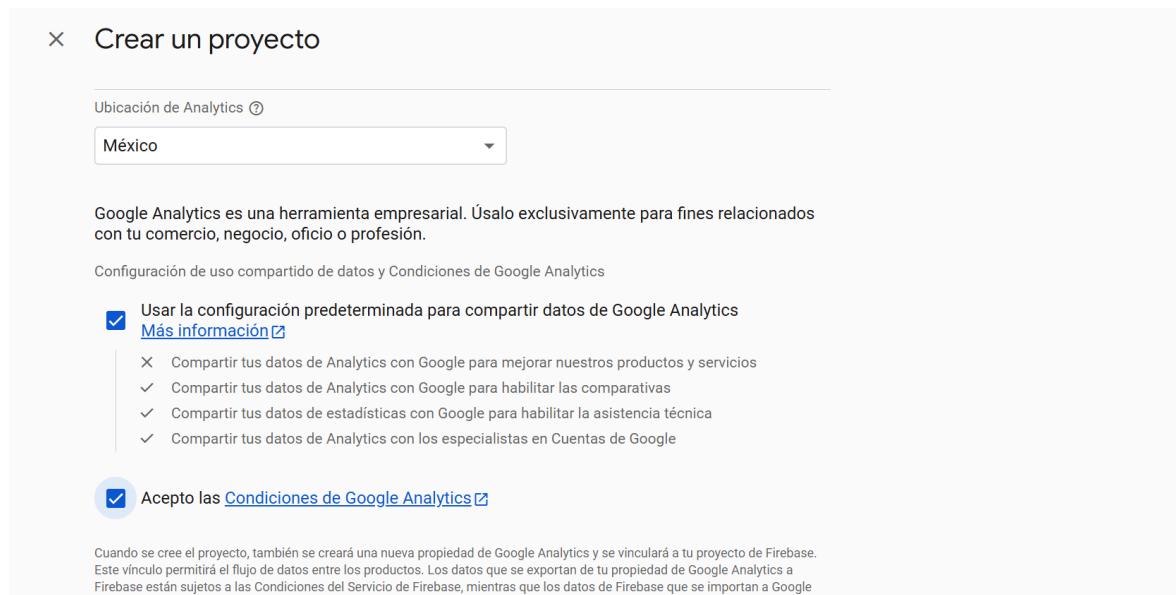


Ilustración 8 Seleccionar País

Una vez realizado lo anterior se procederá a crear el proyecto. El proceso puede tardar un poco.

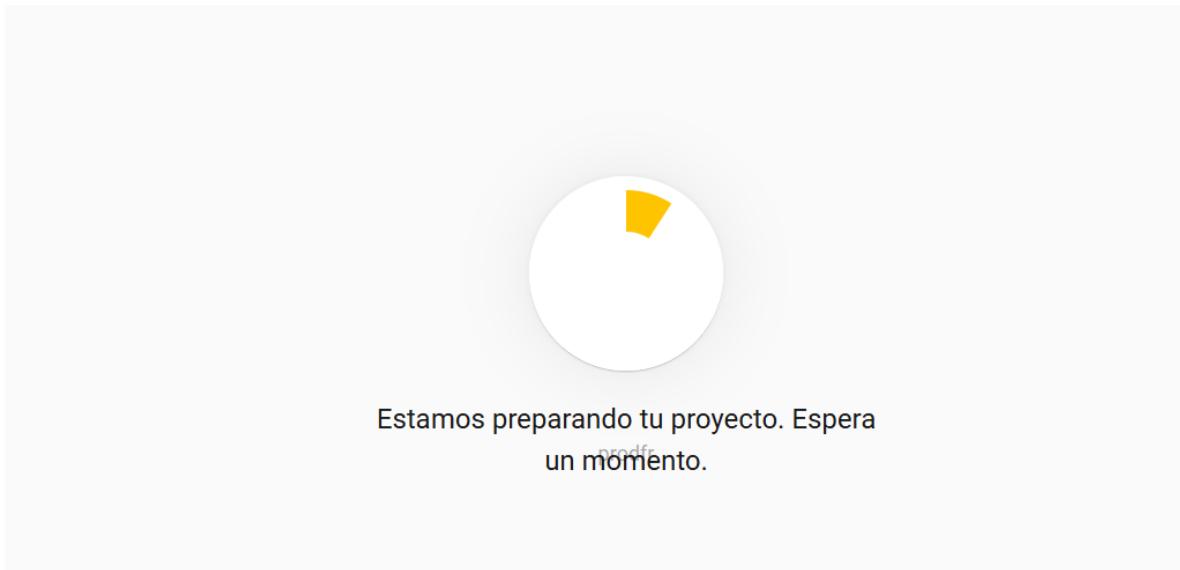


Ilustración 9 Creando Proyecto

HABILITAR AUTENTICACIÓN

Una vez creado el proyecto podremos visualizar la pantalla principal que será el entorno en donde podremos trabajar con nuestro proyecto.

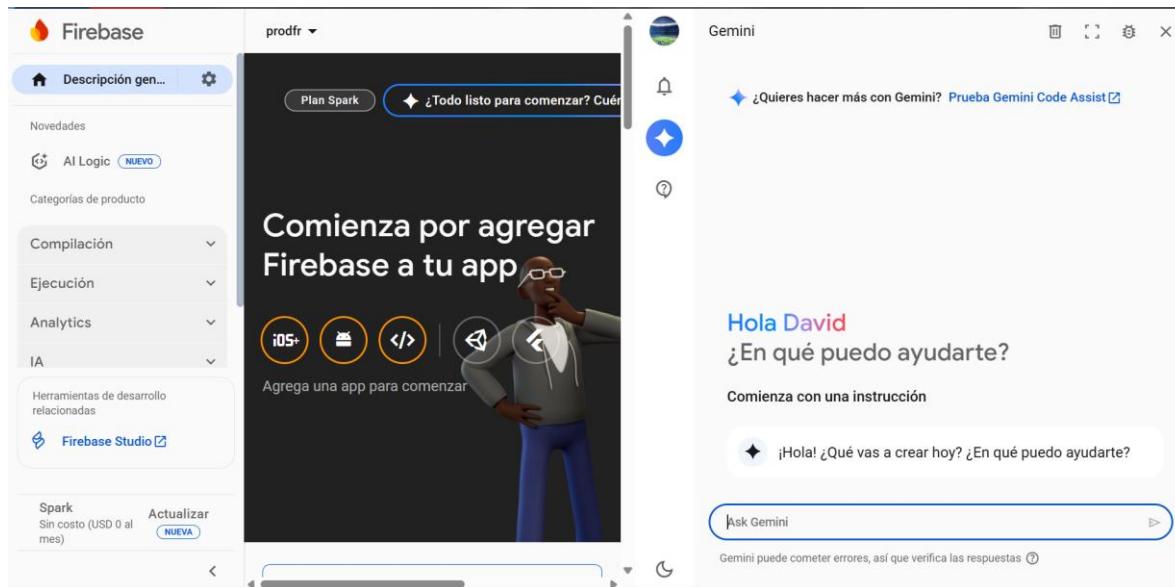


Ilustración 10 Pagina principal del proyecto

Primer paso autenticación:

En el menú izquierdo podremos encontrar distintos apartados nos ubicaremos en la casilla de Authentication el cual nos desplegará una pantalla con la opción comenzar presionaremos comenzar.

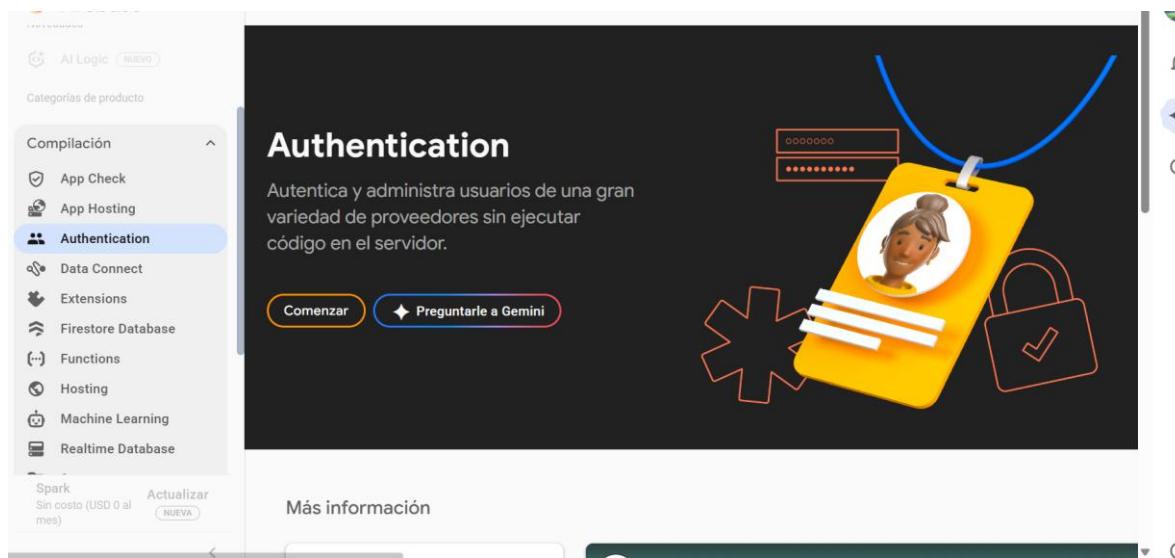


Ilustración 11 Autenticación

Seguido de esto veremos la siguiente pantalla el cual nos pedirá seleccionar un método de acceso para la autenticación de firebase, en este caso seleccionaremos la opción de correo electrónico/contraseña

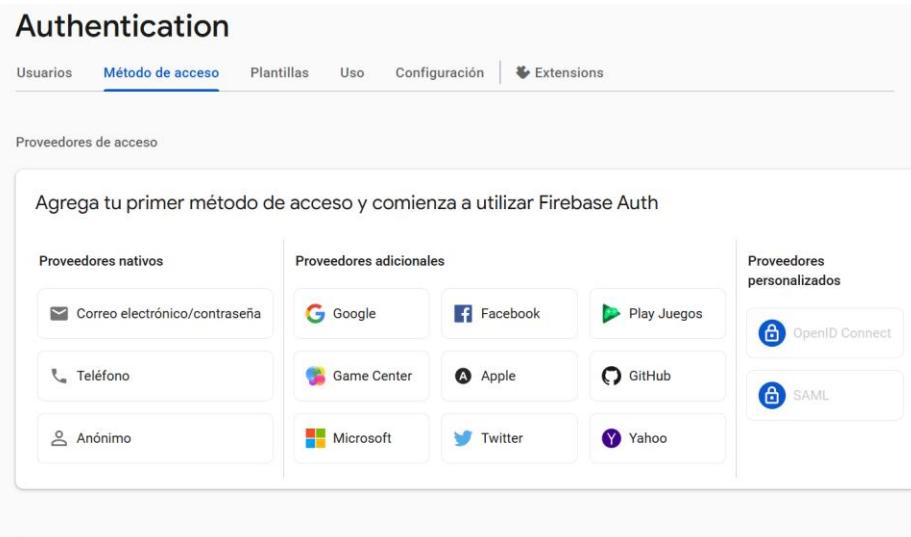


Ilustración 12 Correo y contraseña

Habilitaremos la opción de correo electrónico lo que nos permitirá que el usuario se pueda registrar con su correo electrónico y contraseña.

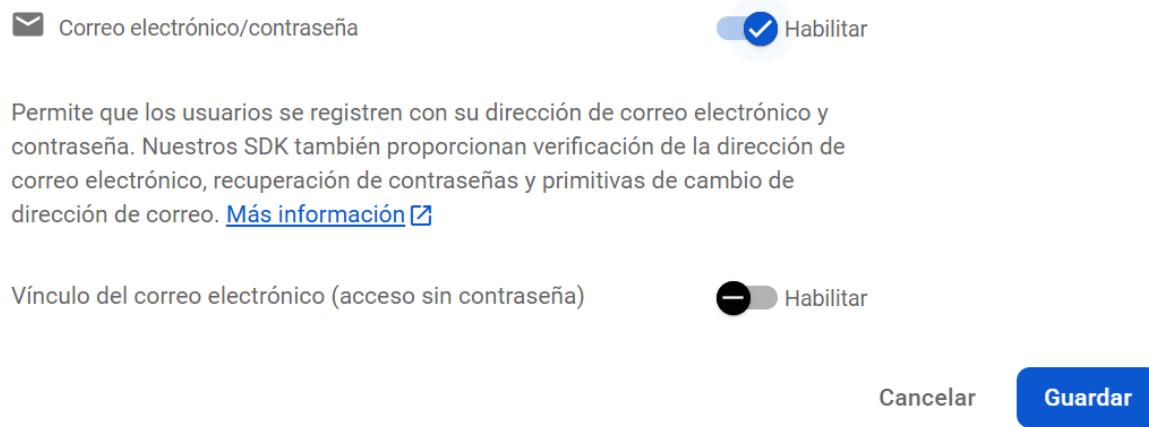


Ilustración 13 Autenticacion Habilitada

Después de haber habilitado la opción nos regresará a la pantalla de métodos de acceso y nos aparecerá los métodos de acceso habilitados.

Proveedores de acceso

Proveedor	Estado
Correo electrónico/contraseña	<input checked="" type="checkbox"/> Habilitada

Agregar proveedor nuevo

Opciones avanzadas

Autenticación de varios factores a través de SMS

Ilustración 14 Auteticacion finalizada

HOSTING

Para empezar con el apartado del hosting nos ubicaremos en la barra izquierda en el apartado de Descripción General > entramos al apartado de configuración de proyecto.

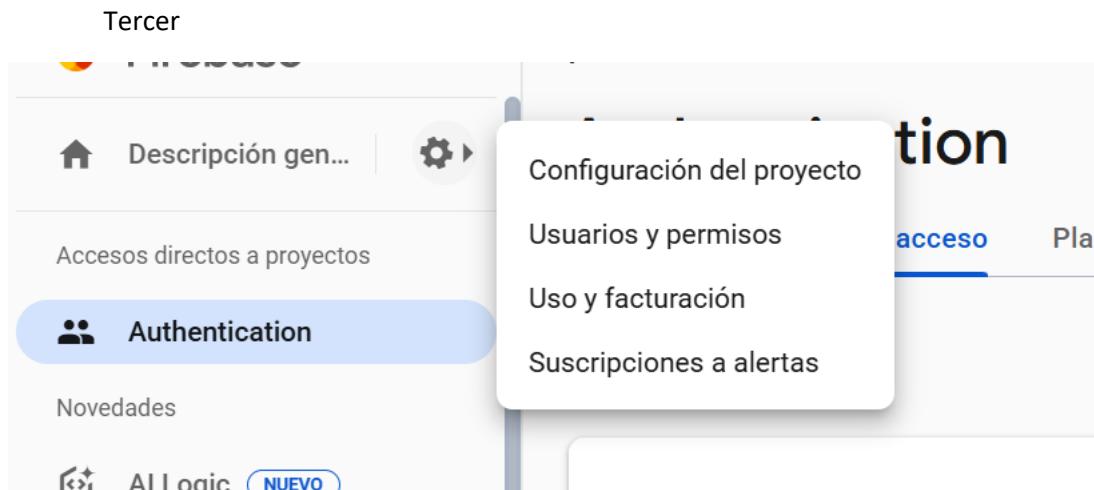


Ilustración 15 Configuracion del proyecto

Dentro nos aparecerá el nombre del proyecto, el id del proyecto y la clave de API web el cual es único y permitirá conectar cualquier app a nuestro proyecto en firebase.

Configuración de proyecto

General Cloud Messaging Integraciones Cuentas de servicio Privacidad de los datos Usuarios y permisos

Tu proyecto

Nombre del proyecto	prodfr
ID del proyecto	prodfr-3baa2
Número del proyecto	1013814917026
Clave de API web	AlzaSyCeRjakmeR-cfkKHbNya4wRCQGQo7z_vZ8

Entorno

Este parámetro de configuración permite personalizar tu proyecto para las diferentes etapas del ciclo de vida de la app

Tipo de entorno Sin especificar

Ilustración 16 Datos del proyecto

Si nos desplazamos hacia abajo nos aparecerá las apps que podemos agregar al proyecto lo cuales pueden ser entornos de ios , andorid y sitios web.

No hay apps en tu proyecto

Selecciona una plataforma para comenzar



Ilustración 17 Apps

En este caso ocuparemos un sitio web por lo que seleccionaremos sitio web, después de ello nos aparecerá el siguiente formulario el cual colocaremos el nombre de la app y habilitaremos el firebase Hosting dejaremos seleccionada la opción que nos manda por default.

1 Registrar app

Sobrenombre de la app ⓘ

Además, configura **Firebase Hosting** para esta app. [Más información ↗](#)

Hosting también se puede configurar más adelante. Puedes comenzar cuando quieras sin costo alguno.

 prodfr-3baa2 (Aún no hay implementaciones) ▾

Registrar app

Ilustración 18 Registro de aplicación

Una vez habilitado el hosting nos aparecerá la siguiente ventana la cual consiste en agregar el SDK de firebase. Seleccionaremos la casilla NPM y abriremos una terminal en nuestra computadora. Escribiremos el siguiente comando: NPM INTALL FIREBASE el cual instalara paquetes de firebase.

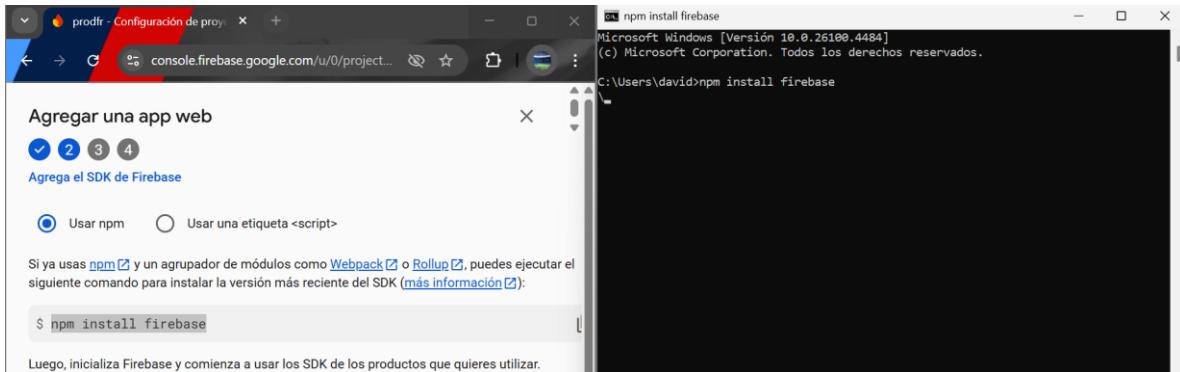


Ilustración 19 Ejecucion del cmd

Como se puede observar la descarga se hará exitosamente

```
C:\Users\david>npm install firebase

added 86 packages in 3m

3 packages are looking for funding
  run `npm fund` for details
npm notice
npm notice New major version of npm available! 10.9.2 -> 11.4.2
npm notice Changelog: https://github.com/npm/cli/releases/tag/v11.4.2
npm notice To update run: npm install -g npm@11.4.2
npm notice

C:\Users\david>
```

Ilustración 20 comando npm install firebase

Siguiendo con las indicaciones colocaremos el siguiente comando `npm install -g firebase-tools`, esto sirve para instalar las herramientas de línea de comandos de Firebase globalmente en tu computadora. Estas herramientas se conocen como Firebase CLI (Command Line Interface).

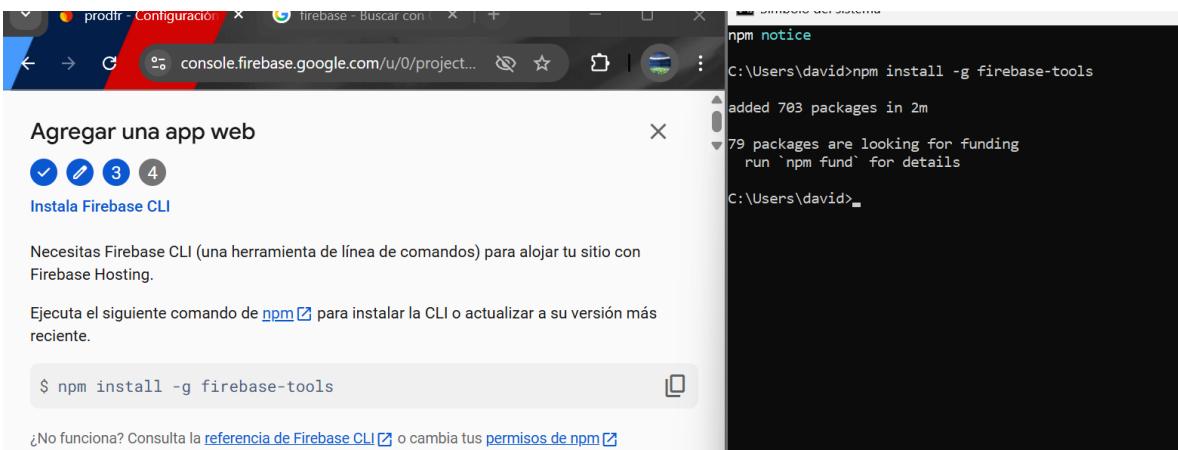


Ilustración 21 comando para instalar las herramientas de firebase

Continuando con la ultima parte de configuración procederemos a logearnos con el correo electrónico registrado ocuparemos el siguiente comando FIREBASE LOGIN

```
C:\Users\david>firebase login  
Already logged in as davidfr339@gmail.com
```

Ilustración 22 Login Exitoso

Nos va a mandar al navegador y deberemos seleccionar la cuenta e iniciar sesión.

 Iniciar sesión con Google

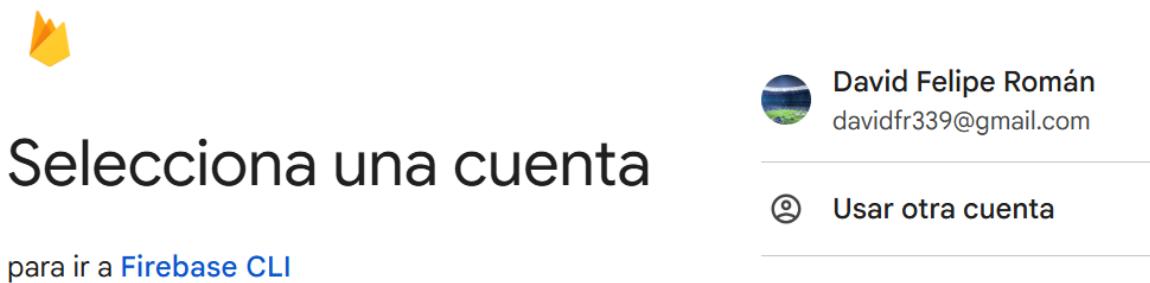


Ilustración 23 Inicio de sesión con correo}

Una vez aceptado los términos y verificar la cuenta nos aparecerá el siguiente mensaje el cual menciona que el login a sido exitoso.

Woohoo!

Firebase CLI Login Successful

You are logged in to the Firebase Command-Line interface. You can immediately close this window and continue using the CLI.

Ilustración 24 inicio de sesión exitoso

Una vez configurado correctamente procederemos a crear una carpeta donde alojaremos el proyecto con el mismo nombre del proyecto de firebase.



Ilustración 25 Carpeta creada

Procedemos a abrir la carpeta en un entorno de desarrollo en este caso será visual studio code dentro de ella crearemos la siguiente estructura

>index.html

>app.js

>css

Este código HTML forma la base de mi sistema de reservación de laboratorios. Aquí se estructura la interfaz principal que incluye el formulario de inicio de sesión, registro y el apartado donde los usuarios pueden hacer una reserva indicando la fecha, hora y laboratorio. También muestra las reservas ya realizadas y permite cerrar sesión. Se conecta con Firebase para manejar la autenticación y la base de datos en la nube. Además, se aplican estilos externos desde un archivo CSS y se utiliza la fuente "Poppins" para darle un mejor diseño. Toda la funcionalidad, como iniciar sesión o reservar, está programada en el archivo app.js.

```
14 <body>
15   <header>
16     | Sistema de Reservas □
17   </header>
18
19   <div class="container">
20     <div id="login-section">
21       <input type="email" id="email" placeholder="Correo electrónico">
22       <input type="password" id="password" placeholder="Contraseña">
23       <button onclick="login()">Iniciar sesión</button>
24       <button onclick="register()">Registrarse</button>
25     </div>
26
27     <div id="reserva-section" style="display: none;">
28       <p id="user-info"></p>
29       <input type="date" id="fecha">
30       <input type="time" id="hora">
31       <input type="text" id="laboratorio" placeholder="Laboratorio (Ej. Lab 3)">
32       <button onclick="reservar()">Reservar</button>
33
34       <h3>Mis Reservas:</h3>
35       <ul id="lista-reservas"></ul>
36       <button onclick="logout()">Cerrar sesión</button>
37     </div>
38   </div>
39
40   <script src="app.js"></script>
41 </body>
42 </html>
```

Ilustración 26 Código HTML

Este script se encarga de la lógica del sistema. Configura Firebase para autenticar usuarios y guardar reservas. Incluye funciones para registrar, iniciar y cerrar sesión, validar datos y evitar reservas en fechas u horas pasadas. También verifica que no haya conflictos con otras reservas del mismo laboratorio. Si todo está bien, guarda la reserva en Firestore y muestra las reservas del usuario actual.

```

2  firebase.initializeApp({
3    apiKey: "AIzaSyCeRjakmeR-cfkKHbNya4wRCQGQo7z_vZ8",
4    authDomain: "prodfr-3baa2.firebaseio.com",
5    projectId: "prodfr-3baa2",
6    storageBucket: "prodfr-3baa2.appspot.com",
7    messagingSenderId: "1013814917026",
8    appId: "1:1013814917026:web:3d593a802c36ee918124fb",
9    measurementId: "G-2RKL2V44ZC"
10   });
11
12 const auth = firebase.auth();
13 const db = firebase.firestore();
14
15 // Bloquear fechas pasadas al cargar la página
16 window.onload = () => {
17   const inputFecha = document.getElementById("fecha");
18   const hoy = new Date().toISOString().split("T")[0];
19   inputFecha.setAttribute("min", hoy);
20 };
21
22 // Validar email
23 function esEmailValido(email) {
24   const regex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;
25   return regex.test(email);
26 }
27
28 // Limpiar campos de login/registro
29 function limpiarCamposAuth() {
30   document.getElementById("email").value = "";
31   document.getElementById("password").value = "";
32 }
33
34 // INICIO DE SESIÓN
35 function login() {
36   const email = document.getElementById("email").value.trim();
37   const password = document.getElementById("password").value.trim();
38
39   if (!email || !password) {
40     alert("Por favor, completa todos los campos.");
41     return;
42   }
43
44   if (!esEmailValido(email)) {
45     alert("Correo electrónico inválido.");
46     return;
47   }
48
49   auth.signInWithEmailAndPassword(email, password)
50     .then(() => {
51       alert("Bienvenido");
52       limpiarCamposAuth();
53     })
54     .catch(e => alert("Error: " + e.message));
55 }
56
57 // CERRAR SESIÓN
58 function logout() {
59   auth.signOut().then(() => alert("Sesión cerrada"));
60 }

```

Ilustración 27 código app.js

Una vez el proyecto este completo procederemos a inicializar firebase en nuestro proyecto. Nos preguntara que si queremos proceder, teclearemos la tecla y para continuar.

```
C:\Users\david>firebase init

#####
##      ##  #####  #####  #####
##      ##  ## ##  ## ##  ##  ##
#####  ##  ## #####  #####  #####  #####
##      ##  ##  ##  ##  ##  ##  ##  ##  ##
##      ##### ##  #####  #####  ##  ##  #####  #####
You're about to initialize a Firebase project in this directory:

C:\Users\david

Before we get started, keep in mind:

* You are initializing your home directory as a Firebase project directory

? Are you ready to proceed? (Y/n) -
```

Ilustración 28 Inicializamos firebase

Configuración Inicial:

El sistema pregunta qué funcionalidades desea activar; en este caso, se selecciona **Hosting** (alojamiento web).

Conexión con Firebase:

Se elige un proyecto existente en la nube de Firebase (llamado "prodfr") para vincularlo con la carpeta local.

Configuración Básica:

Se define la carpeta public como el lugar donde se guardarán los archivos del sitio web.

Se decide **no** convertir el proyecto en una aplicación de una sola página (SPA), lo que significa que el sitio tendrá páginas independientes (como "contacto.html" o "nosotros.html").

```

C:\Users\david

Before we get started, keep in mind:

  * You are initializing your home directory as a Firebase project directory
  ✓ Are you ready to proceed? Yes
  ✓ Which Firebase features do you want to set up for this directory? Press Space to
    select features, then Enter to confirm your choices. Hosting: Configure files for
    Firebase Hosting and (optionally) set up GitHub Action deploys

== Project Setup

First, let's associate this project directory with a Firebase project.
You can create multiple project aliases by running firebase use --add,
but for now we'll just set up a default project.

  ✓ Please select an option: Use an existing project
  ✓ Select a default Firebase project for this directory: prodfr-3baa2 (prodfr)
  i  Using project prodfr-3baa2 (prodfr)

== Hosting Setup

Your public directory is the folder (relative to your project directory) that
will contain Hosting assets to be uploaded with firebase deploy. If you
have a build process for your assets, use your build's output directory.

  ✓ What do you want to use as your public directory? public
  ✓ Configure as a single-page app (rewrite all urls to /index.html)? No
  ✓ Set up automatic builds and deploys with GitHub? No
  + Wrote public/404.html
  + Wrote public/index.html

  + Wrote configuration info to firebase.json
  + Wrote project information to .firebaserc
  + Wrote .gitignore

+ Firebase initialization complete!

```

Ilustración 29 Configuración Inicial

Esto nos creará una carpeta llamada public que contendrá dos carpetas 404 e index.

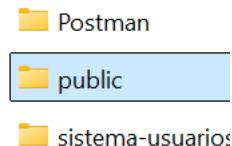


Ilustración 30 Carpeta Public

	Nombre	Fecha de modificación	Tipo
	404	09/07/2025 10:57 a. m.	Chro
	index	09/07/2025 10:57 a. m.	Chro
gas			
rio			
l			

Ilustración 31 Carpetas 404 e index

Una vez las carpetas estén creadas y la configuración se haya realizado correctamente deberemos copiar el archivo app.js, index.html y style.css de la carpeta local que creamos, a la carpeta public alojada en Disco local > Usuarios y debemos reemplazar las mismas.

Una vez realizado el proceso deberemos de ejecutar el comando FIREBASE DEPLOY el cual nos ayudara a subir y publicar nuestro proyecto a Firebase, haciendo que el sitio, funciones o configuraciones estén disponibles en la nube para uso público o pruebas.

Una vez este completo no mandara el link de nuestro sitio.

```
C:\Users\david\Escritorio\prodfr>firebase deploy

== Deploying to 'prodfr-3baa2'...

i  deploying hosting
i  hosting[prodfr-3baa2]: beginning deploy...
i  hosting[prodfr-3baa2]: found 2 files in public
+  hosting[prodfr-3baa2]: file upload complete
i  hosting[prodfr-3baa2]: finalizing version...
+  hosting[prodfr-3baa2]: version finalized
i  hosting[prodfr-3baa2]: releasing new version...
+  hosting[prodfr-3baa2]: release complete

+ Deploy complete!

Project Console: https://console.firebaseio.google.com/project/prodfr-3baa2/overview
Hosting URL: https://prodfr-3baa2.web.app

C:\Users\david\Escritorio\prodfr>
C:\Users\david\Escritorio\prodfr>
```

Ilustración 32 subir proyecto con deploy

Si accedemos a ese link podremos ver el proyecto que estábamos trabajando localmente

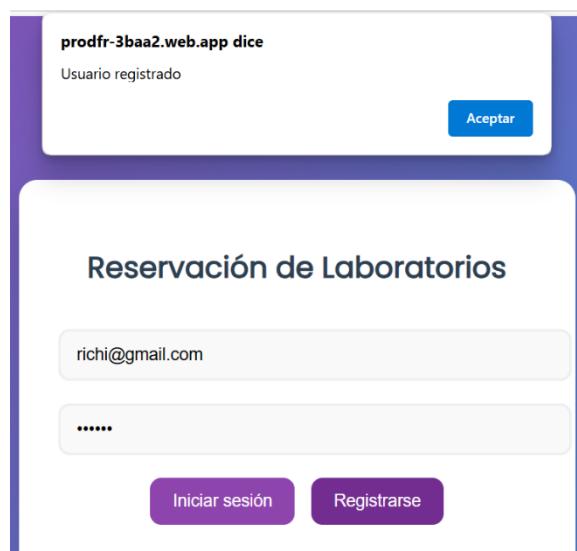
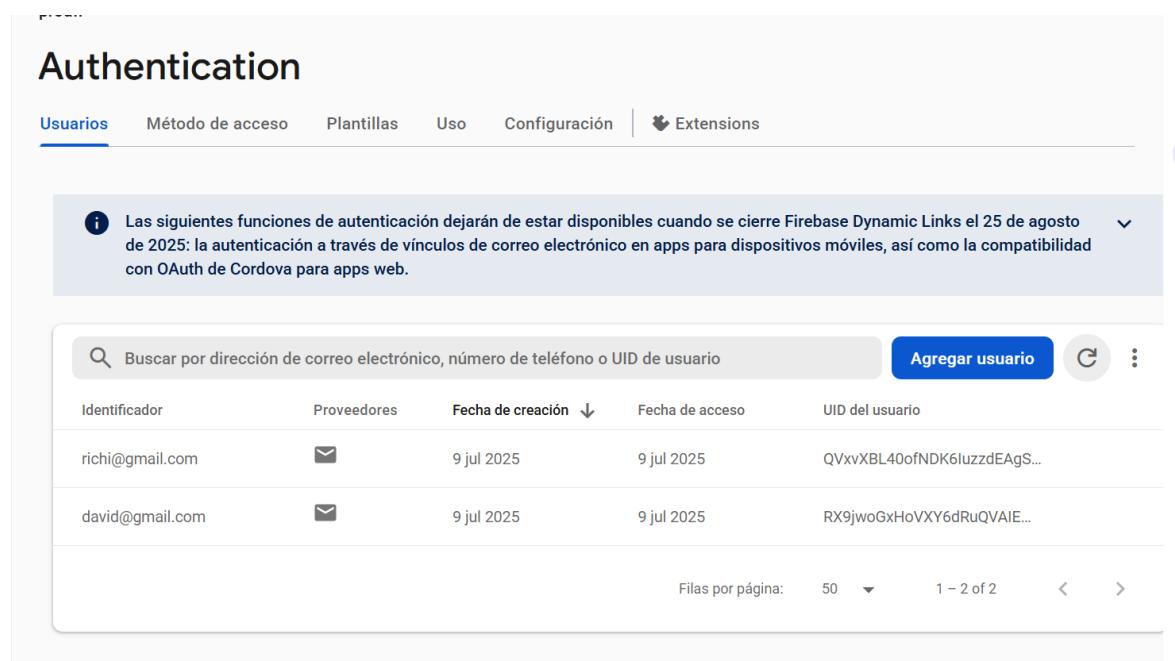


Ilustración 33 inicio de sesión

Y si registramos a un nuevo usuario este deberá de guardarse o registrarse en el apartado de Authentication en el módulo de usuarios.



The screenshot shows the Firebase Authentication console under the 'Users' tab. A message at the top states: 'Las siguientes funciones de autenticación dejarán de estar disponibles cuando se cierre Firebase Dynamic Links el 25 de agosto de 2025: la autenticación a través de vínculos de correo electrónico en apps para dispositivos móviles, así como la compatibilidad con OAuth de Cordova para apps web.' Below this, a search bar and an 'Agregar usuario' button are visible. The user list table contains two entries:

Identificador	Proveedores	Fecha de creación	Fecha de acceso	UID del usuario
richi@gmail.com		9 jul 2025	9 jul 2025	QVxvXBL40ofNDK6luzzdEAgS...
david@gmail.com		9 jul 2025	9 jul 2025	RX9jwoGxHoVXY6dRuQVAIE...

At the bottom, there are pagination controls: 'Filas por página: 50' and '1 - 2 of 2'.

Ilustración 34 Usuarios Guardados

HABILITAR BASES DE DATOS

Para que nuestro sitio web cumpla con su propósito es evidente que deberá contar con una base de datos. Firebase proporciona la herramienta de base de datos así que habilitaremos esta opción.

Nos ubicamos nuevamente en la barra izquierda y desplegamos la opción de compilación y dentro de ella habrá un apartado llamado firestore database. Al presionarlo nos saldrá la siguiente ventana en la cual deberemos seleccionar la opción Crear base de datos

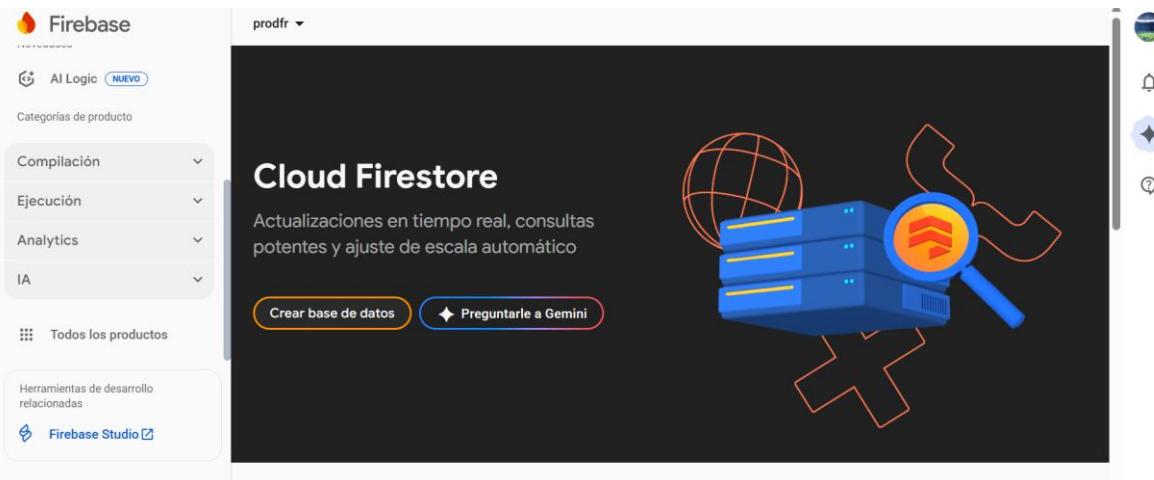


Ilustración 35 crear base de datos

En esta sección del asistente de creación de la base de datos en Firebase, se le solicita al usuario definir las reglas de seguridad para el acceso a Firestore. Se presentan dos opciones:

- Modo de producción: Restringe el acceso a la base de datos. Solo se permite la lectura y escritura conforme a las reglas personalizadas que se definan.
- Modo de prueba : Permite el acceso total (lectura y escritura) a cualquier usuario con la referencia de la base de datos, por un período de 30 días. Después de este tiempo, será necesario actualizar las reglas para mantener el acceso.

Seleccionaremos de momento el modo de prueba.

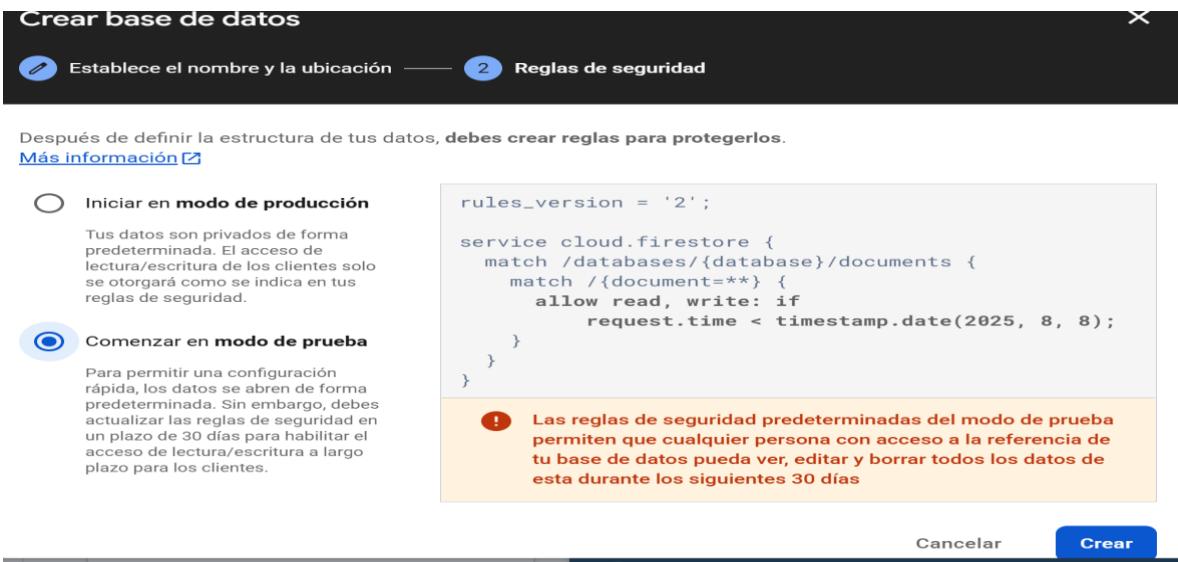


Ilustración 36 Reglas de Seguridad

Una vez realizado lo anterior pulsmao la opción de crear. Y no habrá creado nuestra base de datos lista para usarse. En este caso el proyecto realizado crea por si mi

Ilustración 37 Crear colección

En este caso el proyecto realizado crea por si mismo la colección donde se guardaran todos los registros de reservaciones que se realicen en el sitio y que como podemos ver se guardan exitosamente en la base creada anteriormente.

The screenshot shows the Google Cloud Firestore interface. At the top, there are two notifications: one about a security alert (facturación o phishing) and another about the introduction of the Enterprise edition of Firestore compatible with MongoDB. There are buttons for 'Más información' and 'Ignorar'. Below the notifications, there are tabs for 'Vista del panel' (selected), 'Compilador de consultas', and a more options menu.

The main area displays a hierarchical navigation path: Home > reservas > tAvp620AMLek... . To the right, there is a link to 'Más funciones en Google Cloud'.

The data is presented in a table format:

(default)	reservas	tAvp620AMLekQ5k13Fe0
+ Iniciar colección	+ Agregar documento	+ Iniciar colección
reservas >	tAvp620AMLekQ5k13Fe0 >	+ Agregar campo
		email: "richi@gmail.com"
		fecha: "2025-07-11"
		hora: "13:04"
		laboratorio: "Lab2"
		uid: "QVxvXBL40ofNDK6luzzdEAgShmr2"

Ilustración 38 Datos guardados en la base de datos

VALIDACIONES

Para que el usuario no pueda registrar reservaciones vacías o duplicadas debemos validar los distintos campos.

1.- Campos Vacíos:

Debemos validar los campos para evitar campos vacíos para así no permitir el registro de reservaciones que no cumplan con los datos solicitados

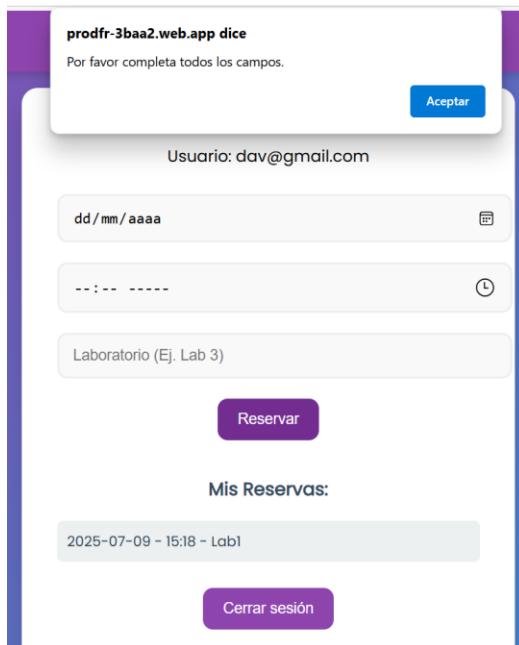


Ilustración 39 Validacion campos vacíos

```
...if(!fecha || !hora || !lab === ""){  
...  alert("Por favor completa todos los campos.");  
...  return;  
...}
```

Ilustración 40 código de validación

2.-Fechas Anteriores.

Para que no haya confusión y errores en los datos el sistema deberá evitar registrar reservaciones anteriores a la fecha actual.

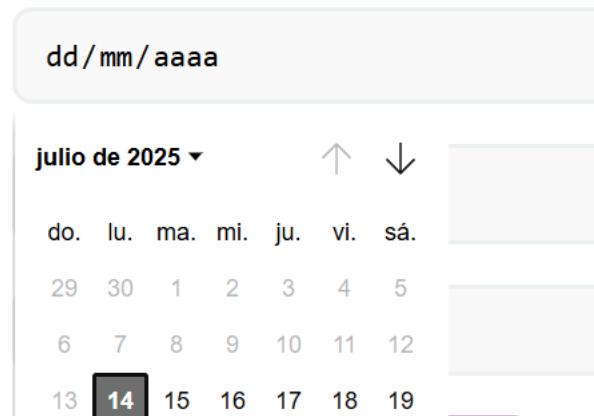


Ilustración 41 Validacion de día

```
const ahora = new Date();
const hoy = ahora.toISOString().split("T")[0];

if(fecha < hoy) {
    alert("No puedes reservar para una fecha pasada.");
    return;
}
```

Ilustración 42 Código de validación

3.- Horas cruzadas

Por último para que un laboratorio no pueda ser seleccionado mientras haya una reserva realizada. Se aplica un rango de 2 horas para q el usuario no pueda registrar una reserva

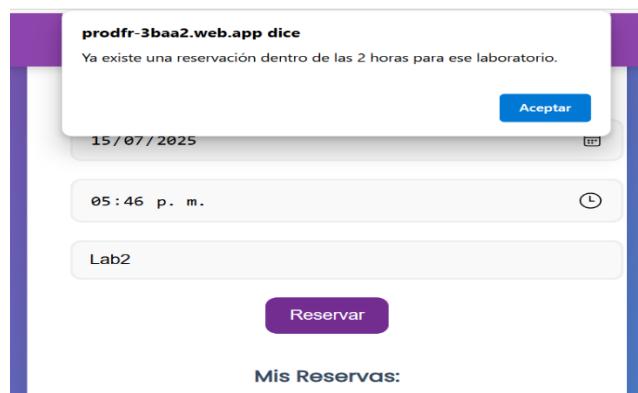


Ilustración 43 Validacion de reserva

dentro del lapso mencionado.

```
...//·Convertir·hora·a·número·de·minutos·para·comparar·rangos
...const [h, ·m] = ·hora.split(":").map(Number);
...const minutosSeleccionados = h * 60 + m;

...db.collection("reservas")
...··where("fecha", "·==", ·fecha)
...··where("laboratorio", "·==", ·lab)
...·get()
...·then(snapshot => {
...··let conflicto = false;

...··snapshot.forEach(doc => {
...···const reserva = doc.data();
...···const [hr, min] = reserva.hora.split(":").map(Number);
...···const minutosExistente = hr * 60 + min;

...····const diferencia = Math.abs(minutosSeleccionados - -minutosExistente);
...····if (diferencia < 120) { //·menos·de·2·horas
...·····conflicto = true;
...···}
...·});
...··if (conflicto) {
...···alert("Ya existe una reservación dentro de las 2 horas para ese laboratorio.");
...···return;
...·};

...};
```

Ilustración 44 Código de validacion

ESTILOS

Como pagina principal tenemos el login el cual el usuario podrá iniciar sesión con un usuario creado previamente o podrá registrar uno nuevo, para ello deberá de colocar una dirección de correo electrónico y alguna contraseña mayor a seis dígitos.

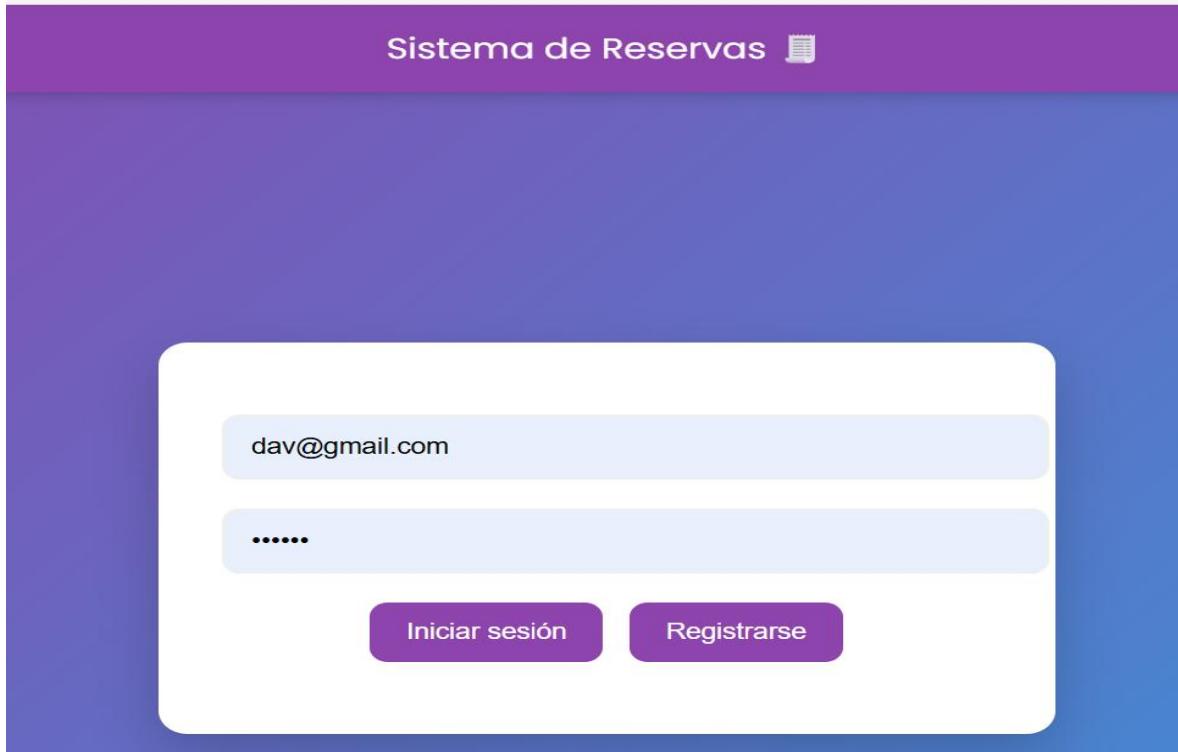


Ilustración 45 Inicio de sesión con estilos

Después de que el usuario inicie sesión correctamente podrá visualizar el sistema de reservas el cual le permitirá escoger la fecha y hora de la reservación así como del laboratorio a reservar.

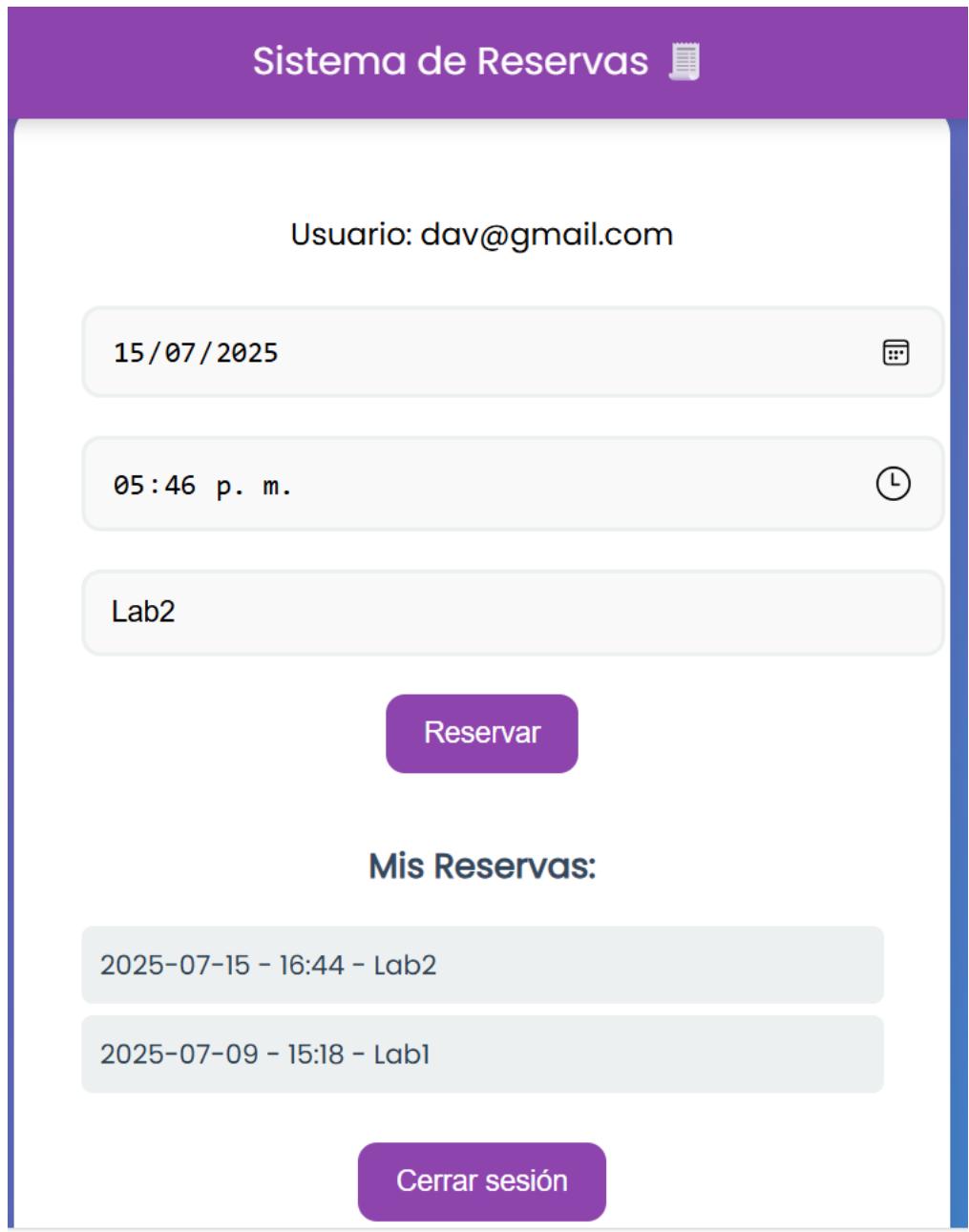


Ilustración 46 Pagina de reservaciones con estilos

También podrá visualizar las reservas que haya realizado anteriormente.

El campo fecha ayudara al usuario a poder seleccionar la fecha en la que reservara algún laboratorio

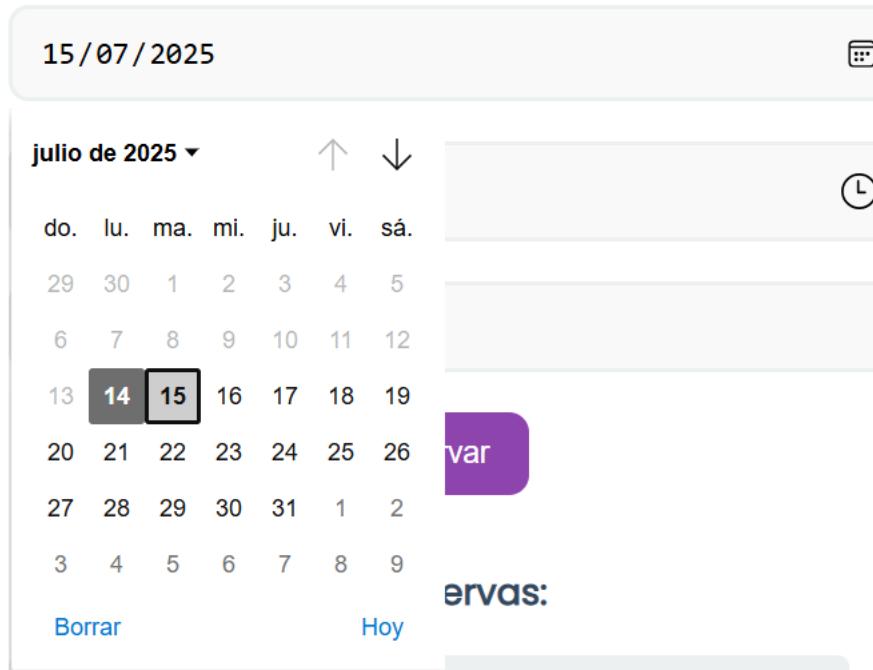


Ilustración 47 Formato de Fecha

El campo hora servirá para seleccionar el horario el cual el usuario requiera el laboratorio.

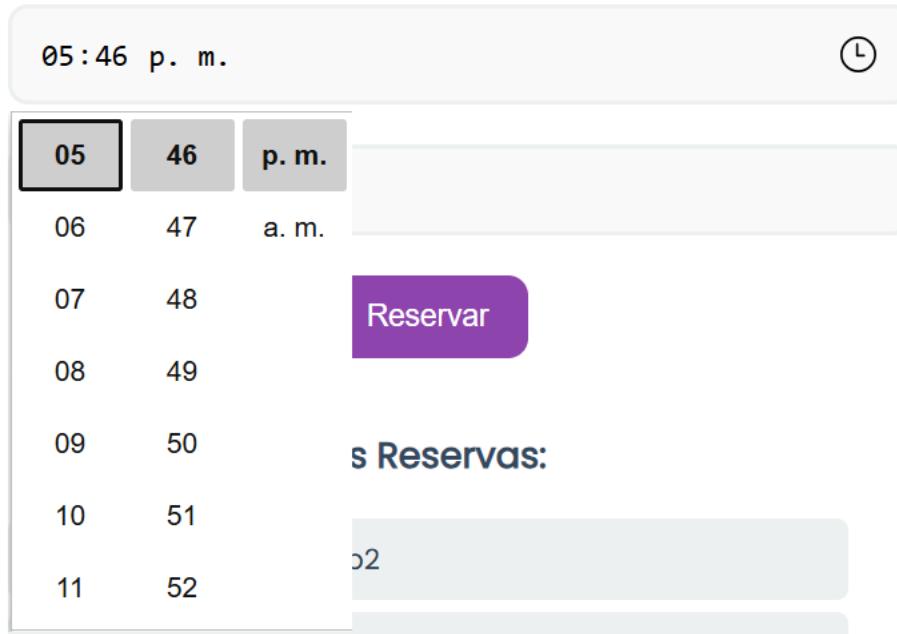


Ilustración 48 Formato de Hora

Análisis

En este proyecto se utilizó Firebase para manejar la autenticación de usuarios y el almacenamiento de las reservas en la nube. Se hizo la configuración para que solo los usuarios registrados puedan acceder y realizar reservaciones, además de implementar validaciones para evitar reservas en fechas u horas pasadas y conflictos en el mismo laboratorio.

Se diseñó una interfaz sencilla que permite a los usuarios hacer y consultar sus reservas fácilmente. Para esto, se usaron tecnologías web como HTML, CSS y JavaScript junto con Firebase, lo que hace que el sistema sea accesible desde cualquier dispositivo con internet y fácil de mantener.

Durante el desarrollo aprendí a utilizar lo básico de Firebase, como la autenticación con correo y contraseña, el manejo de la base de datos en Firestore y la integración con aplicaciones web. Esto me ayudó a comprender cómo funcionan los servicios en la nube y cómo aplicarlos en proyectos prácticos. En general, se logró crear un sistema funcional y práctico que facilita la administración de las reservas en los laboratorios.



TECNOLÓGICO
NACIONAL DE MÉXICO

TESSFP
TECNOLÓGICO DE ESTUDIOS SUPERIORES
SAN FELIPE DEL PROGRESO

TECNOLÓGICO DE ESTUDIOS SUPERIORES DE SAN FELIPE DEL PROGRESO

CÓMPUTO EN LA NUBE

Segundas Prácticas

DOCENTE:

ING. LIZETH JOSE ANGELES

ALUMNO:

DAVID FELIPE ROMAN

INGENIERÍA INFORMÁTICA

NOVENO SEMESTRE

INDICE

.....	6
PRACTICA1	6
INTRODUCCIÓN.....	6
REQUERIMIENTOS	6
CREACIÓN DE CUENTA EN AZURE Y GITHUB	7
REGISTRO O INICIO DE SESION GITHUB.....	14
DESCARGAR GITHUB DESKTOP	17
PROYECTO EN STATIC WEB APPS	25
VINCULAR AZURE – GITHUB	27
CREACIÓN DE DNS	33
VINCULAR AZURE - FIREBASE.....	39
ANÁLISIS DE LA PRÁCTICA	43
PRACTICA 2	44
INTRODUCCIÓN	44
REQUISITOS GENERALES	44
CREACION DE MAQUINA VIRTUAL	45
ALMACENAMIENTO.....	68
PRACTICA 3	79
INTRODUCCIÓN:	79
Requisitos:.....	79
CONEXIÓN REMOTA Y TRANSFERENCIA DE ACHIVOS A MV DE AZURE	80
ANALISIS	91
 Ilustración 1 Navegación hacia "Precios" en el sitio de Azure. Autoría propia	7
Ilustración 2 Seleccionamos "Azure for Students	8
Ilustración 3 Registro en Azure para estudiantes. Autoría propia	8
Ilustración 4 Formulario de datos personales y escolares. Autoría propia	9
Ilustración 5 Proceso de confirmación de cuenta. Autoría propia	9
Ilustración 6 Aceptamos condiciones y presionamos "Registrarse". Autoría propia	10
Ilustración 7 Instalación de Microsoft Authenticator. Autoría propia	10
Ilustración 8 Ingreso del número de teléfono para verificación. Autoría propia	11
Ilustración 9 Teléfono verificado exitosamente. Autoría propia	12
Ilustración 10 Cargando Azure. Autoría propia.....	12
Ilustración 11 Panel principal de Azure Education. Autoría propia	13
Ilustración 12 Servicios incluidos con la cuenta de estudiante. Autoría propia	13
Ilustración 13 Búsqueda de GitHub en Google. Autoría propia	14
Ilustración 14 Vista del panel si ya teníamos cuenta. Autoría propia.....	14

Ilustración 15 Formulario para crear un repositorio en GitHub. Autoría propia	15
Ilustración 16 Configuración final del repositorio. Autoría propia.....	16
Ilustración 17 Opciones para subir archivos al repositorio. Autoría propia	16
Ilustración 18 Página de descarga de GitHub Desktop. Autoría propia	17
Ilustración 19 Bienvenida de GitHub Desktop. Autoría propia	18
Ilustración 20 Confirmamos la cuenta para GitHub Desktop. Autoría propia	18
Ilustración 21 Permitir acceso de GitHub Desktop a nuestra cuenta. Autoría propia	19
Ilustración 22 Configurar nombre y correo para los commits. Autoría propia	19
Ilustración 23 Crear repositorio local en GitHub Desktop. Autoría propia.....	20
Ilustración 24 Crear carpeta local para el proyecto. Autoría propia	20
Ilustración 25 Confirmar ubicación y crear repositorio local. Autoría propia	21
Ilustración 26 Visualización de repositorios en GitHub. Autoría propia.....	22
Ilustración 27 Archivos del proyecto dentro del repositorio local. Autoría propia	22
Ilustración 28 Subir archivos con un commit en GitHub Desktop. Autoría propia	23
Ilustración 29 Opción para publicar el repositorio en GitHub. Autoría propia.....	23
Ilustración 30 Publicar repositorio desde GitHub Desktop. Autoría propia.....	24
Ilustración 31 Repositorio publicado en GitHub con archivos cargados. Autoría propia ...	24
Ilustración 32 Búsqueda del servicio “Static Web Apps” en Azure. Autoría propia	25
Ilustración 33 Vista inicial de Static Web Apps en Azure. Autoría propia.....	25
Ilustración 34 Formulario para crear una Static Web App. Autoría propia	26
Ilustración 35 Creación del grupo de recursos “AzureDFR” . Autoría propia	26
Ilustración 36 Selección de repositorio GitHub en Azure. Autoría propia.....	27
Ilustración 37 Configuración de compilación y flujo de trabajo. Autoría propia	27
Ilustración 38 Validación final. Autoría propia	28
Ilustración 39 Inicialización del despliegue. Autoría propia.....	28
Ilustración 40 Implementación correcta. Autoría propia.....	29
Ilustración 41 Ver aplicación en el explorador. Autoría propia.....	29
Ilustración 42 Sistema de Reservas - Pantalla de inicio. Autoría propia.....	30
Ilustración 43 cambio en la aplicación. Autoría propia.....	31
Ilustración 44 Confirmación del tercer compromiso en GitHub. Autoría propia.....	31
Ilustración 45 Actualización de la aplicación web en Azure. Autoría propia.....	32
Ilustración 46 selección de laboratorio. Autoría propia	32
Ilustración 47 Búsqueda de servicio para la creación de Zona DNS en Azure. Autoría propia	33
Ilustración 48 Administrando el sitio en Firebase. Autoría propia	34
Ilustración 49 Agregar un dominio personalizado. Autoría propia.....	34
Ilustración 50 Escribimos nuestro dominio personalizado. Autoría propia	35
Ilustración 51 Ya se agregó el dominio, ahora toca configurar el DNS. Autoría propia....	36
Ilustración 52 zona DNS en Azure. Autoría propia	37
Ilustración 53 Confirmamos la zona DNS. Autoría propia	37
Ilustración 54 Esperamos a que se cree la zona. Autoría propia.....	38
Ilustración 55 Zona DNS creada con éxito. Autoría propia	38
Ilustración 56 agregar los registros DNS. Autoría propia.....	39
Ilustración 57 Agregar Registros. Autoría propia.....	39
Ilustración 58 Agregamos el registro tipo A para apuntar el dominio a Firebase. Autoría propria	40

Ilustración 59 Agregamos el registro TXT para validar el dominio en Firebase. Autoría propia	41
Ilustración 60 Consulta de propagación DNS para el dominio www.mindsdavidfr.com. Autoría propia	42
<i>Ilustración 61 Entrar a PUTTYGEN.</i> Autoría propia	45
<i>Ilustración 62 Generación de aleatoriedad moviendo el cursor.</i> Autoría propia.....	46
<i>Ilustración 63 Visualización y guardado de la clave pública y privada.</i> Autoría propia.....	47
<i>Ilustración 64 Asignación de passphrase para mayor seguridad.</i> Autoría propia.....	47
<i>Ilustración 65 Ventana para guardar el archivo de clave privada.</i> Autoría propia	48
<i>Ilustración 66 Nombrar y guardar el archivo .ppk generado.</i> Autoría propia.....	48
<i>Ilustración 67 Archivo de clave privada almacenado en el escritorio.</i> Autoría propia.....	49
<i>Ilustración 68 Seleccionar “Crear un recurso” desde el portal de Azure</i>	49
<i>Ilustración 69 Elegir “Máquina virtual” desde servicios populares.</i> Autoría propia	50
<i>Ilustración 70 Formulario inicial para crear una máquina virtual en Azure.</i> Autoría propia 51	
<i>Ilustración 71 Asignar nombre, región y zona de disponibilidad a la VM</i>	52
<i>Ilustración 72 Elegir Ubuntu como sistema operativo para la VM.</i> Autoría propia.....	52
<i>Ilustración 73 Asignar usuario y método de autenticación a la VM.</i> Autoría propia.....	53
<i>Ilustración 74 Paso 14: Ingresar una clave pública SSH existente.</i> Autoría propia.....	53
<i>Ilustración 75 Seleccionar opciones de almacenamiento para el sistema operativo.</i> Autoría propia	54
<i>Ilustración 76 Seleccionar tipo de almacenamiento para el sistema operativo.</i> Autoría propia	55
<i>Ilustración 77 Definir red virtual, subred e IP pública.</i> Autoría propia	56
<i>Ilustración 78 Verificación de información y advertencia sobre puertos abiertos.</i> Autoría propia	57
<i>Ilustración 79 La implementación de la máquina virtual está en curso.</i> Autoría propia....	58
<i>Ilustración 80 Se completó correctamente la creación de la máquina virtual.</i> Autoría propia	58
<i>Ilustración 81 Información general de la máquina virtual activa.</i> Autoría propia	59
<i>Ilustración 82 Seleccionar archivo de clave privada para la conexión.</i> Autoría propia.....	60
<i>Ilustración 83 Clave privada seleccionada para autenticación SSH.</i> Autoría propia	61
<i>Ilustración 84 Ingresar el nombre de usuario para la conexión SSH.</i> Autoría propia	62
<i>Ilustración 85 Ingresar IP pública y guardar sesión SSH.</i> Autoría propia.....	63
<i>Ilustración 86 Sesión SSH guardada con nombre personalizado.</i> Autoría propia.....	63
<i>Ilustración 87 Conexión establecida: autenticación mediante clave SSH.</i> Autoría propia	64
<i>Ilustración 88 Sesión iniciada en Ubuntu desde PuTTY.</i> Autoría propia.....	65
<i>Ilustración 89 Actualización del sistema y reinicio de servicios.</i> Autoría propia	66
<i>Ilustración 90 Creación de una carpeta desde la terminal.</i> Autoría propia	67
<i>Ilustración 91 Alta de usuario y asignación de directorio propio.</i> Autoría propia	67
<i>Ilustración 92 Alta de grupo en el sistema.</i> Autoría propia	67
<i>Ilustración 93 Ubicación del servicio “Cuentas de almacenamiento” en Azure.</i> Autoría propia	68
<i>Ilustración 94 Pantalla inicial del servicio “Cuentas de almacenamiento” .</i> Autoría propia 69	
<i>Ilustración 95 Formulario de configuración de cuenta de almacenamiento en Azure.</i> Autoría propia	69
<i>Ilustración 96 Selección de región, rendimiento y redundancia.</i> Autoría propia.....	70

<i>Ilustración 97 Resumen de configuración antes de crear la cuenta de almacenamiento.</i>	
Autoría propia	71
<i>Ilustración 98 Inicializando la implementación de la cuenta de almacenamiento. Autoría propia</i>	71
<i>Ilustración 99 Implementación en curso en Azure. Autoría propia.....</i>	72
<i>Ilustración 100 Implementación completada correctamente. Autoría propia.....</i>	73
<i>Ilustración 101 Vista general del recurso creado. Autoría propia</i>	73
<i>Ilustración 102 Cargar archivos al blob. Autoría propia</i>	74
<i>Ilustración 103 Crear nuevo contenedor. Autoría propia</i>	74
<i>Ilustración 104 Confirmación de contenedor creado. Autoría propia</i>	75
<i>Ilustración 105 Archivo seleccionado para carga. Autoría propia</i>	76
<i>Ilustración 106 Confirmación de carga exitosa. Autoría propia.....</i>	76
<i>Ilustración 107 Navegación a Contenedores. Autoría propia.....</i>	77
<i>Ilustración 108 Visualización del archivo cargado en el contenedor. Autoría propia.....</i>	77
<i>Ilustración 109 Propiedades del archivo cargado. Autoría propia.....</i>	78
<i>Ilustración 110 Página oficial de descarga de WinSCP. Autoría propia.....</i>	80
<i>Ilustración 111 Inicio automático de la descarga de WinSCP. Autoría propia</i>	81
<i>Ilustración 112 Acuerdo de Licencia de WinSCP. Autoría propia</i>	81
<i>Ilustración 113 Selección del tipo de instalación en WinSCP. Autoría propia</i>	82
<i>Ilustración 114 Selección de apariencia inicial en WinSCP. Autoría propia</i>	83
<i>Ilustración 115 Resumen final antes de instalar WinSCP. Autoría propia.....</i>	84
<i>Ilustración 116 Instalación en progreso de WinSCP. Autoría propia</i>	84
<i>Ilustración 117 Importar sesiones almacenadas de PuTTY a WinSCP. Autoría propia</i>	85
<i>Ilustración 118 Selección de sitios para importar desde PuTTY. Autoría propia.....</i>	86
<i>Ilustración 119 Finalización de la instalación de WinSCP. Autoría propia</i>	87
<i>Ilustración 120 Configuración de conexión en WinSCP a la máquina virtual de Azure.</i>	
Autoría propia	88
<i>Ilustración 121 Autenticación con clave pública en WinSCP. Autoría propia</i>	88
<i>Ilustración 122 Interfaz de WinSCP: Conexión establecida con la máquina virtual. Autoría propia</i>	89
<i>Ilustración 123 Subir archivo desde la máquina local a la máquina virtual. Autoría propia89</i>	
<i>Ilustración 124 Confirmación de la transferencia exitosa de archivo en WinSCP. Autoría propia</i>	90
<i>Ilustración 125 Confirmación de archivo transferido en la terminal de Ubuntu. Autoría propia</i>	91

PRACTICA1

INTRODUCCIÓN

En esta práctica se realizó una integración completa de servicios en la nube utilizando tres plataformas clave: Microsoft Azure, GitHub y Firebase. El objetivo fue desplegar un sitio web mediante Azure Static Web Apps, conectarlo directamente con un repositorio en GitHub para automatizar el despliegue continuo, y finalmente configurar un dominio personalizado vinculado a los servidores de Firebase para gestionar el hosting DNS.

Este tipo de configuración representa un ejemplo claro de cómo implementar entornos modernos, escalables y automatizados utilizando herramientas gratuitas o educativas disponibles para estudiantes y desarrolladores.

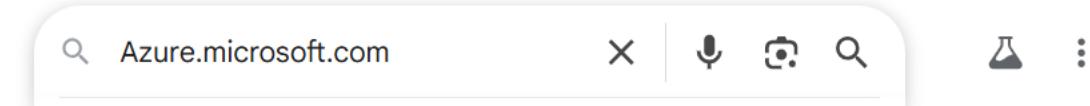
REQUERIMIENTOS

Para llevar a cabo esta práctica fue necesario contar con lo siguiente:

- Cuenta activa en **Microsoft Azure** con suscripción educativa (*Azure for Students*).
- Cuenta funcional en **GitHub**, con un repositorio público o privado del proyecto web.
- Acceso a **Firebase Console** para gestión de hosting y configuración DNS.
- Navegador web actualizado (Chrome, Edge o Firefox).
- Conexión a internet estable.
- Herramientas complementarias:
 - Visual Studio Code (opcional para edición de archivos web).
 - Cuenta de correo electrónico institucional para el alta en plataformas.

CREACIÓN DE CUENTA EN AZURE Y GITHUB

1.- Para empezar con la práctica debemos ir al navegador y buscar la página oficial de azure.microsoft.com.



2.- Una vez dentro de la página principal de Azure (<https://azure.microsoft.com>), nos dirigimos a la barra superior de navegación. Haz clic en el menú "Más" y selecciona la opción "Precios". Esto te mostrará diferentes opciones relacionadas con costos y servicios gratuitos de Azure.

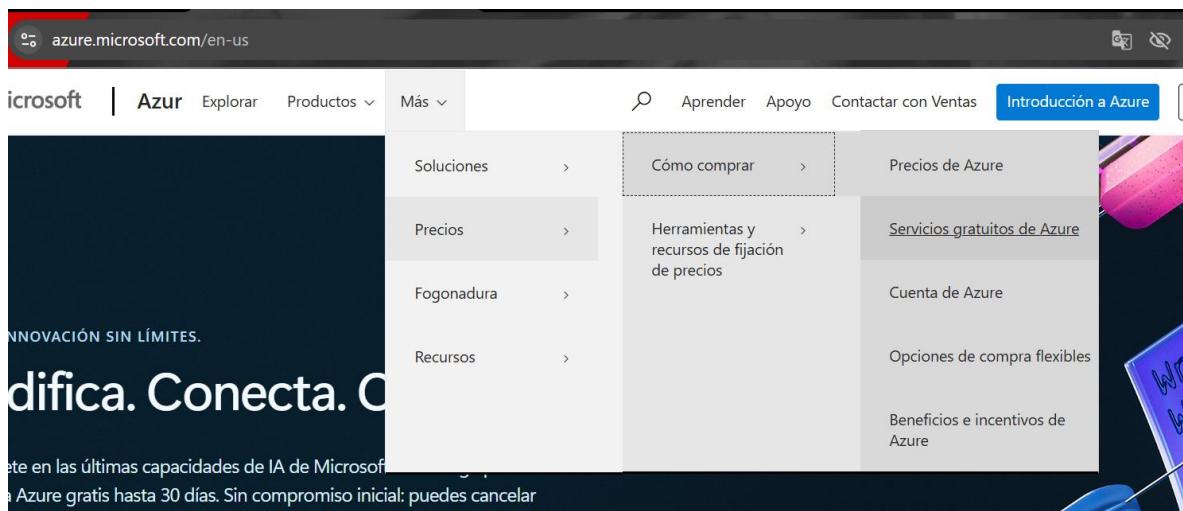


Ilustración 1 Navegación hacia "Precios" en el sitio de Azure. Autoría propia

3.- En esta sección, elegimos la opción "Azure for Students" para obtener herramientas gratuitas y \$100 USD en crédito usando solo nuestro correo académico.

Other ways to start with Azure

Pay as you go

Get started with pay-as-you-go pricing and only pay for what you use beyond monthly free amounts of services. No up-front commitment. Cancel anytime.

[Learn more](#)



Azure for Students

Get access to free developer tools and \$100 credit. No credit card required—all you need is an academic email address.

[Learn more](#)

Ilustración 2 Seleccionamos "Azure for Students"

4.-Presionamos el botón "Empieza gratis" para registrarnos con nuestro correo universitario y obtener los \$100 USD sin necesidad de tarjeta.

Construya en la nube gratis con Azure para estudiantes

Utiliza tu correo electrónico de universidad o escuela para registrarte y renovar cada año que seas estudiante

[Empieza gratis](#)

[Obtenga más información sobre la elegibilidad](#)

Comience con
\$100 de
crédito de
Azure

No se
requiere
tarjeta de
crédito

Ilustración 3 Registro en Azure para estudiantes. Autoría propia

5.-Llenamos el formulario con nuestro nombre completo, país, nombre de la escuela y fecha de nacimiento según los registros escolares.

The screenshot shows a web-based form for entering personal and educational information. At the top, there is a descriptive text box. Below it, there are input fields for Name (Nombre) containing "David", Surname (Apellidos) containing "Felipe Roman", and Country/Region (País o región) set to "México". A note below the country field states: "Si su país no aparece en la lista, la oferta no está disponible en su región. [Más información](#)". There is also a field for School Name (Nombre de la escuela) with the placeholder "Escribir el nombre de una escuela". Below this, a note says: "El nombre de la escuela le ayudará a proporcionar a Microsoft información adicional para la verificación. Si está disponible, intodúzcalo aquí.". Finally, there is a Date of Birth (Fecha de nacimiento) field showing "08/10/2003" with a calendar icon.

Ilustración 4 Formulario de datos personales y escolares. Autoría propia

5.- Despues de ingresar nuestros datos, el sistema comienza a confirmar automáticamente la cuenta para validar la información registrada.

Confirmando su cuenta...

Ilustración 5 Proceso de confirmación de cuenta. Autoría propia

6.- Marcamos las casillas de verificación y luego hacemos clic en "Registrarse" para completar el proceso de activación de la cuenta de Azure para estudiantes.

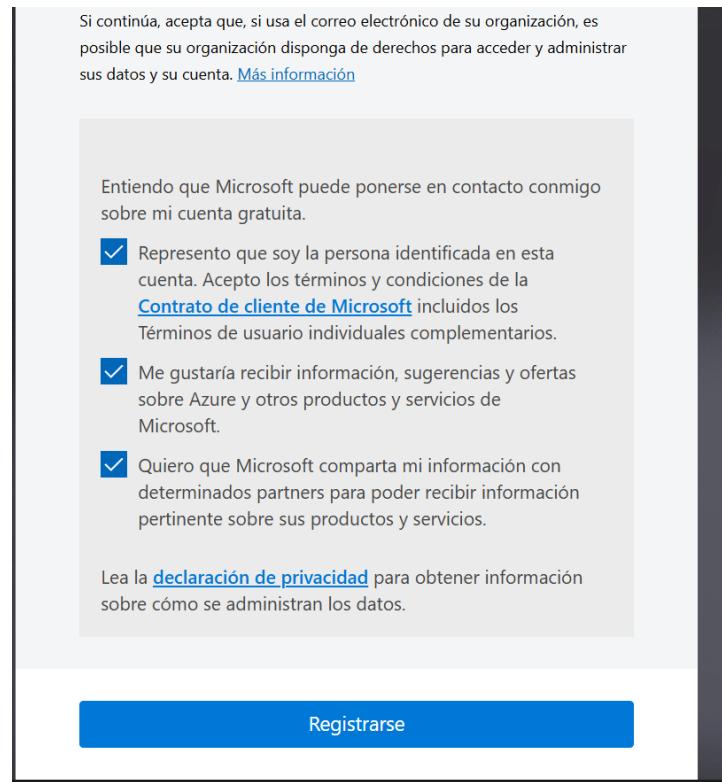


Ilustración 6 Aceptamos condiciones y presionamos "Registrarse". Autoría propia

7.- Para proteger la cuenta, instalamos la app Microsoft Authenticator en el celular. Luego presionamos "Siguiente" para continuar con la verificación.

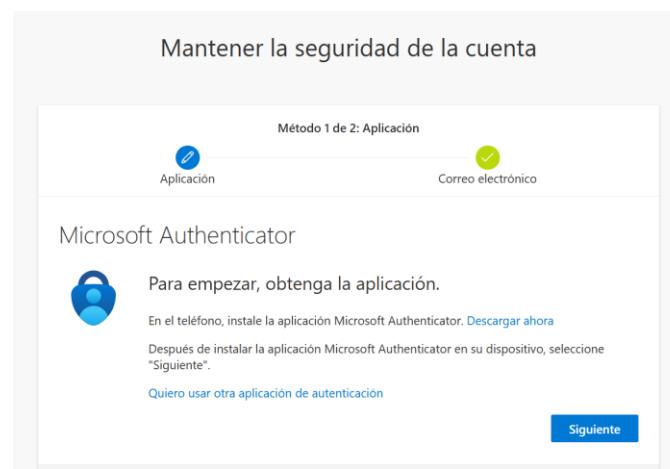


Ilustración 7 Instalación de Microsoft Authenticator. Autoría propia

8.- Ingresamos nuestro número de teléfono, seleccionamos la opción "Recibir un código" y hacemos clic en "Siguiente" para completar la verificación por SMS.

Teléfono	Correo electrónico
----------	--------------------

Teléfono

Para verificar tu identidad, puedes optar por recibir un código en tu teléfono.

¿Qué número de teléfono quiere usar?

Código del país	Número de teléfono
Mexico (+52)	7122413990

Elija cómo comprobar

Recibir un código

Se pueden aplicar tarifas de datos y mensajes. Si eliges Siguiente, se aceptan los [Términos del servicio](#) y [Declaración de privacidad y cookies](#).

Siguiente

Ilustración 8 Ingreso del número de teléfono para verificación. Autoría propia

9.- El sistema confirma que el número se registró correctamente. Hacemos clic en "Siguiente" para continuar con la configuración.



Ilustración 9 Teléfono verificado exitosamente. Autoría propia

Después de completar los pasos anteriores, el sistema carga automáticamente el portal de Azure, donde podremos comenzar a usar nuestras herramientas de estudiante.



Ilustración 10 Cargando Azure. Autoría propia

11.- Entramos al portal de Azure con nuestra cuenta de estudiante y confirmamos que tenemos \$100 USD de crédito disponible, válidos por 365 días para usar en recursos y servicios de Microsoft Azure.

The screenshot shows the Microsoft Azure Education Information General page. It displays the following information:

- Detalles de la oferta para estudiantes:**
 - Créditos disponibles: **USD 100 de USD 100**
 - Días hasta que expira el crédito: **365** (Expira el 15/07/2026)
 - Costos de Julio: **USD 0.00**
- Soluciones populares:**
 - Implementación de un contenedor de Docker
 - Cree su primera aplicación node.js
 - Cree y entrene un modelo de Machine Learning
 - Compile e implemente su primer sitio web
- Servicios gratuitos:** Includes links to Machine Learning, Node.js, Docker, and Site Web.
- Software libre:** Includes links to Docker, Node.js, Machine Learning, and Site Web.

Ilustración 11 Panel principal de Azure Education. Autoría propia

12.- Desde el panel, podemos acceder a servicios gratuitos mensuales durante 12 meses, como máquinas virtuales, almacenamiento, bases de datos y herramientas de seguridad. Aquí elegimos lo que deseamos usar según nuestras necesidades.

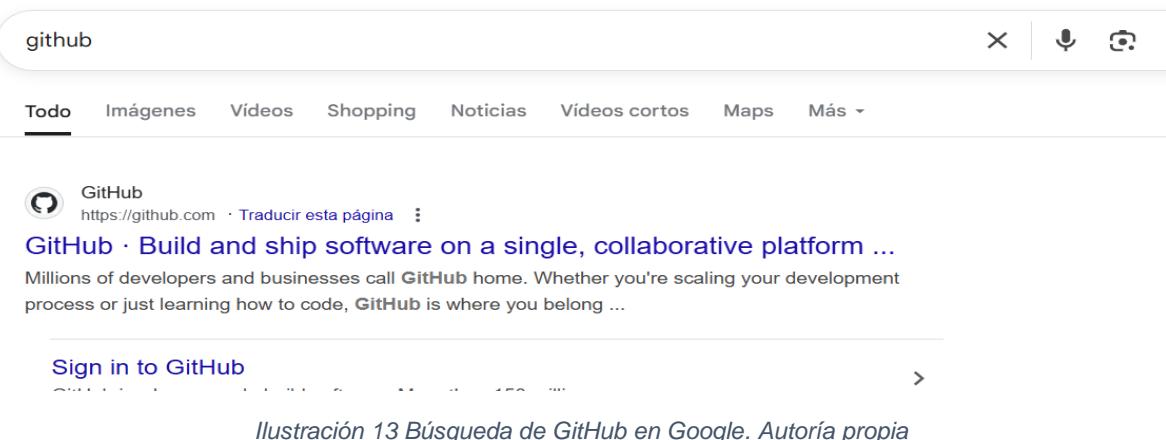
The screenshot shows the Azure Free Services page. It lists the following services and their monthly quotas:

Servicio	Quota Mensual	Descripción
Máquina virtual Windows PROCESO	750 horas	cada una de las máquinas virtuales ampliables B1s y B2ats v2 (basadas en AMD). Cree máquinas virtuales (VM) Windows en cuestión de segundos para satisfacer sus necesidades de carga de trabajo y presupuesto. Más información
Máquina virtual con Linux PROCESO	750 horas	cada una de las máquinas virtuales ampliables B1s y B2ats v2 (basadas en AMD). Cree máquinas virtuales (VM) Linux en cuestión de segundos para satisfacer sus necesidades de carga de trabajo y presupuesto. Más información
Azure Managed Disks ALMACENAMIENTO	64 GB x 2	almacenamiento en discos SSD (unidad de estado sólido) (P6) más 1 GB de instantáneas y 2 millones de operaciones de E/S
Azure Blob Storage ALMACENAMIENTO	5 GB	almacenamiento con redundancia local (LRS) en bloques de acceso frecuente, con 20 000 operaciones de lectura y 10 000 de escritura
Azure Files ALMACENAMIENTO		
Key Vault SEGURIDAD		
Codificación Azure Media Serv... MULTIMEDIA		
Azure Database for MySQL BASES DE DATOS		

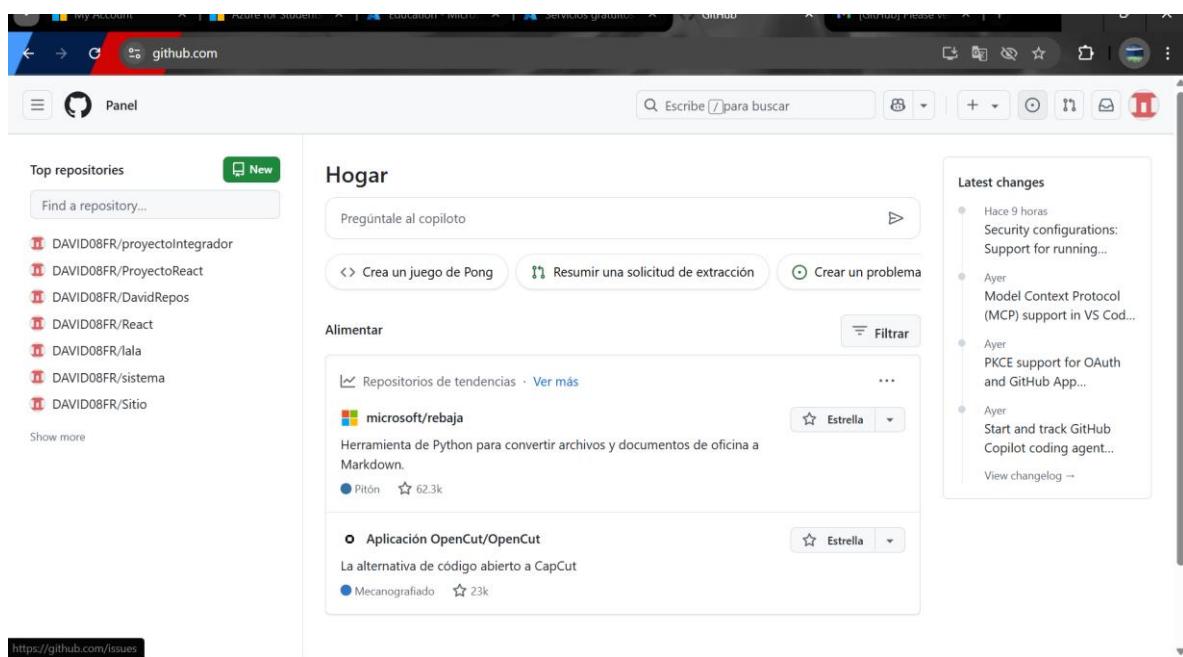
Ilustración 12 Servicios incluidos con la cuenta de estudiante. Autoría propia

REGISTRO O INICIO DE SESION GITHUB

1.- Abrimos el navegador, escribimos “GitHub” en Google y seleccionamos el primer enlace oficial: <https://github.com>, para acceder a la plataforma.



2.- Si ya teníamos una cuenta creada, GitHub nos llevará directamente al panel principal, donde podemos ver nuestros repositorios, proyectos recientes y herramientas como Copilot.



3.- Hacemos clic en el botón "+" en la esquina superior derecha y seleccionamos "Nuevo repositorio". Luego, elegimos el nombre del repositorio, una descripción si lo deseamos, y configuramos si será público o privado.

The screenshot shows the GitHub interface for creating a new repository. At the top, there's a search bar with placeholder text 'Escribe para buscar' and several navigation icons. Below the header, the title 'Crear un nuevo repositorio' is displayed, along with a link to 'Prueba la nueva experiencia'. A note below the title says 'Un repositorio contiene todos los archivos del proyecto, incluido el historial de revisiones. ¿Ya tienes un repositorio de proyecto en otro lugar? [Importa un repositorio](#)'. A note at the bottom of this section states 'Los campos obligatorios están marcados con un asterisco (*)'. The main form area starts with 'Dueño *' followed by a dropdown menu showing 'DAVID08FR' and a slash separator. Next is 'Nombre del repositorio *' with an empty input field. Below these fields is a note: 'Los buenos nombres de repositorios son cortos y fáciles de recordar. ¿Necesitas inspiración? ¿Qué te parece? paraguas potencial?'. There's also an optional 'Descripción (opcional)' input field with a note: 'Inicialice este repositorio con:'. Underneath this, there are two radio button options: 'Público' (selected) and 'Privado'. The 'Público' option is described as 'Cualquier persona en internet puede ver este repositorio. Tú decides quién puede contribuir.' The 'Privado' option is described as 'Tú eliges quién puede ver y comprometerse con este repositorio.' At the bottom of the form, there's a note: 'Asegúrate de escribir una descripción detallada de tu proyecto. Obtén más información sobre los archivos README.'

Ilustración 15 Formulario para crear un repositorio en GitHub. Autoría propia

4.-Escribimos el nombre del repositorio (por ejemplo, "PRODFR"), elegimos si será público o privado, y opcionalmente marcamos la opción para agregar un archivo README, un archivo .gitignore y una licencia. Luego, hacemos clic en el botón "Crear repositorio" al final.

Dueño *

DAVID08FR / PRODFR

PRODFR está disponible.

Los buenos nombres de repositorios son cortos y fáciles de recordar. ¿Necesitas inspiración? ¿Qué te parece? **paraguas potencial** ?

Descripción (opcional)

 Público
Cualquier persona en internet puede ver este repositorio. Tú decides quién puede contribuir.

 Privado
Tú eliges quién puede ver y comprometerse con este repositorio.

Inicialice este repositorio con:

Agregar un archivo README
Aquí puedes escribir una descripción detallada de tu proyecto. [Obtén más información sobre los archivos README](#).

Agregar .gitignore

Plantilla .gitignore : Ninguna

Seleccione los archivos que no desea rastrear de una lista de plantillas. [Obtenga más información sobre cómo ignorar archivos](#).

Elija una licencia

Licencia : Ninguna

Ilustración 16 Configuración final del repositorio. Autoría propia

5.-Después de crear el repositorio, GitHub nos muestra opciones para configurarlo desde el escritorio o usar la línea de comandos. Aquí podemos copiar el enlace del repositorio y seguir las instrucciones para subir nuestros archivos de forma rápida.

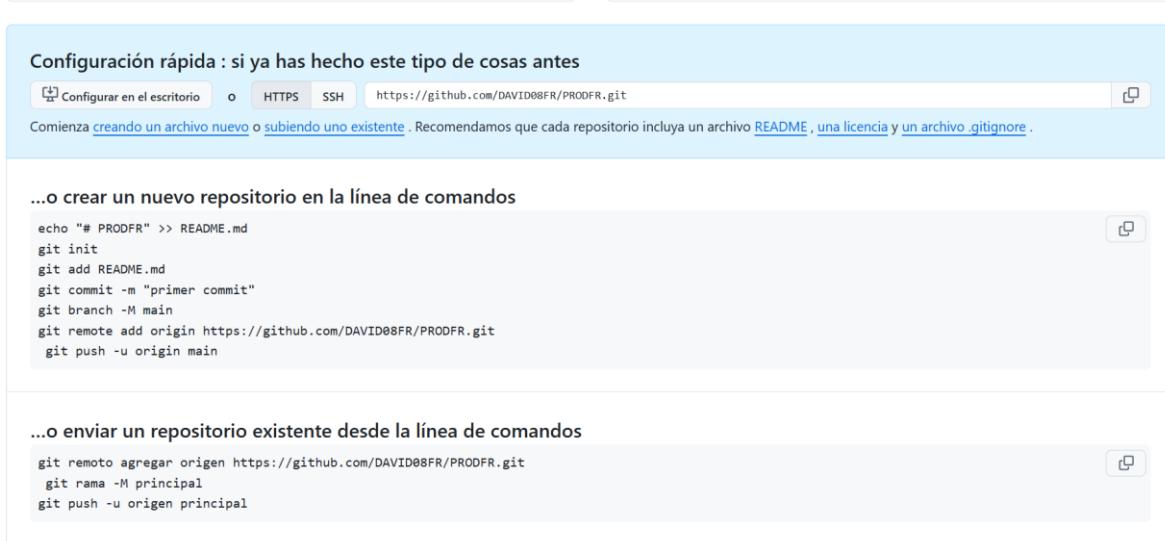


Ilustración 17 Opciones para subir archivos al repositorio. Autoría propia

DESCARGAR GITHUB DESKTOP

1.-Accedemos a <https://desktop.github.com/download/> y hacemos clic en “Descargar para Windows (64 bits)” para instalar la aplicación oficial de GitHub, la cual facilita la gestión de repositorios desde el escritorio sin usar comandos.



Ilustración 18 Página de descarga de GitHub Desktop. Autoría propia

2.-Una vez instalado GitHub Desktop, se abre una ventana donde hacemos clic en “Sign in to GitHub.com” para iniciar sesión y vincular nuestra cuenta. El navegador se encargará de redirigirnos automáticamente a la aplicación.

Welcome to GitHub Desktop

Your browser will redirect you back to GitHub Desktop once you've signed in.
If your browser asks for your permission to launch GitHub Desktop please
allow it to.

[Sign in to GitHub.com](#) [Cancel](#)

New to GitHub? [Create your free account.](#)

Ilustración 19 Bienvenida de GitHub Desktop. Autoría propia

3.- Seleccionamos nuestro usuario (en este caso DAVID08FR) y presionamos “Continuar” para autorizar la conexión entre la cuenta de GitHub y la aplicación de escritorio.

Seleccionar usuario para
autorizar **GitHub Desktop**



Ilustración 20 Confirmamos la cuenta para GitHub Desktop. Autoría propia

4.- Revisamos los permisos solicitados por GitHub Desktop (acceso a repositorios, datos del usuario y flujo de trabajo) y hacemos clic en "Autorizar el escritorio" para completar la conexión entre nuestra cuenta y la aplicación.

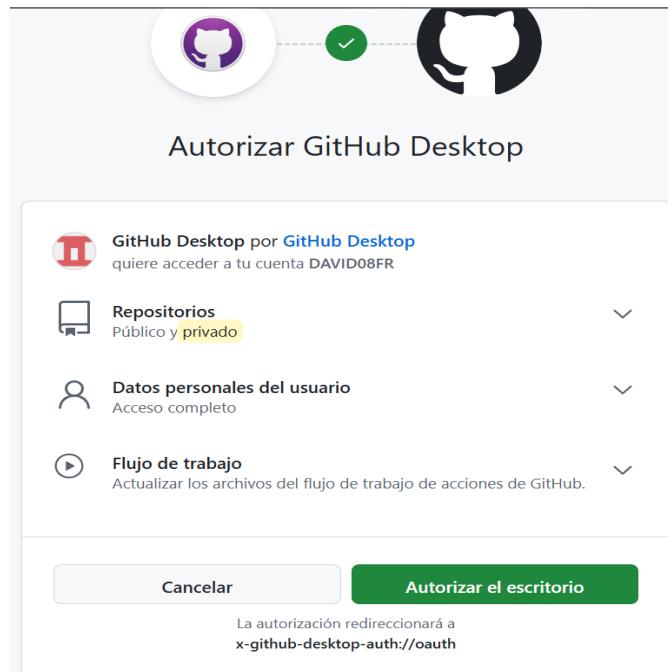


Ilustración 21 Permitir acceso de GitHub Desktop a nuestra cuenta. Autoría propia

5.-Confirmamos que se utilice nuestro nombre de usuario y correo de GitHub para identificar los cambios que subamos. Luego hacemos clic en “Finish” para completar la configuración.

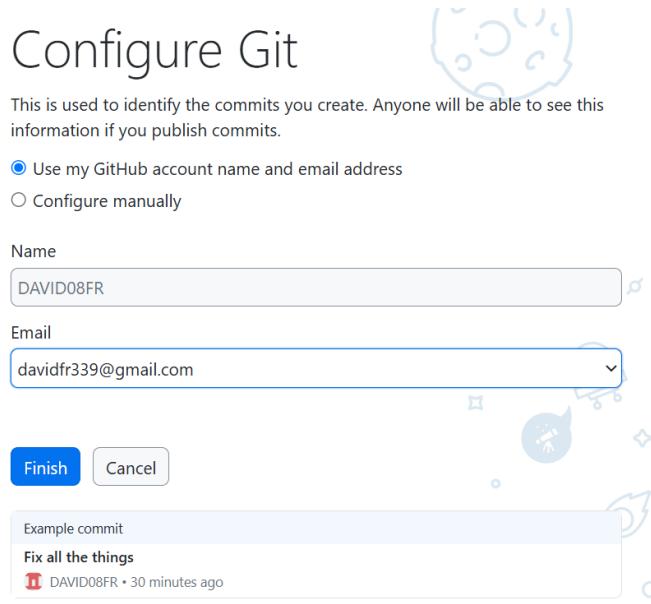


Ilustración 22 Configurar nombre y correo para los commits. Autoría propia

6.-En GitHub Desktop, seleccionamos “Create a new repository”, escribimos el nombre, elegimos una ruta local en nuestro equipo y hacemos clic en “Create repository”. Esto nos permite trabajar localmente antes de subir el proyecto a GitHub.

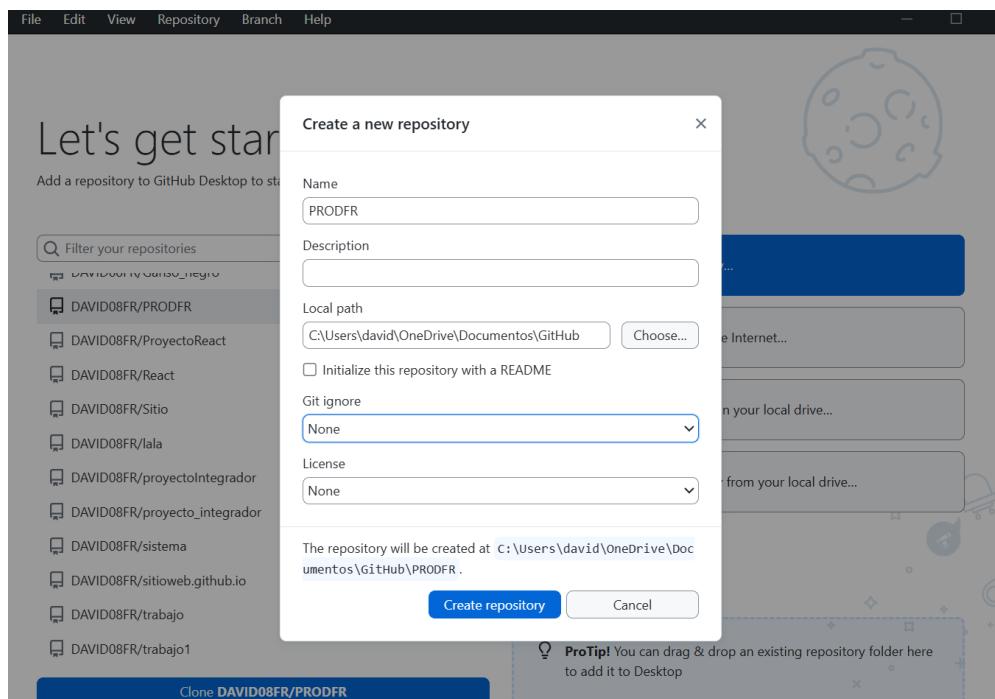


Ilustración 23 Crear repositorio local en GitHub Desktop. Autoría propia

7.-Creamos una nueva carpeta en nuestro equipo (por ejemplo, “PRO”) y la seleccionamos como destino para guardar el repositorio local en GitHub Desktop.

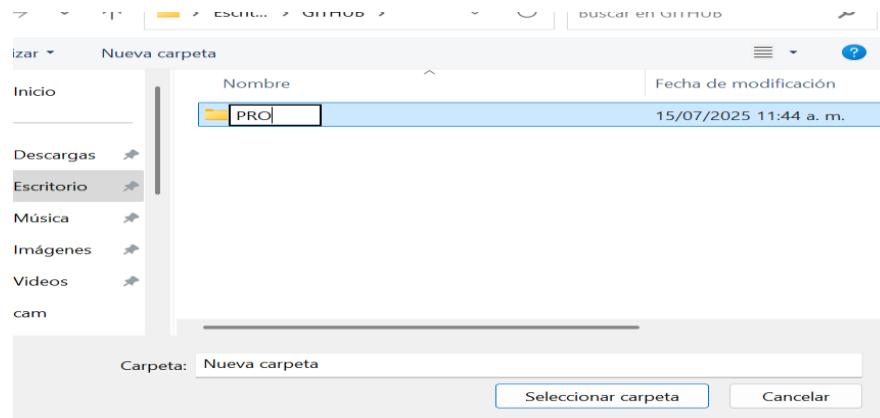


Ilustración 24 Crear carpeta local para el proyecto. Autoría propia

8.-Verificamos que el repositorio se creará en la carpeta seleccionada (por ejemplo, C:\Users\david\Escritorio\GITHUB\PRO\PRODFR) y hacemos clic en “Create repository” para comenzar a trabajar desde GitHub Desktop.

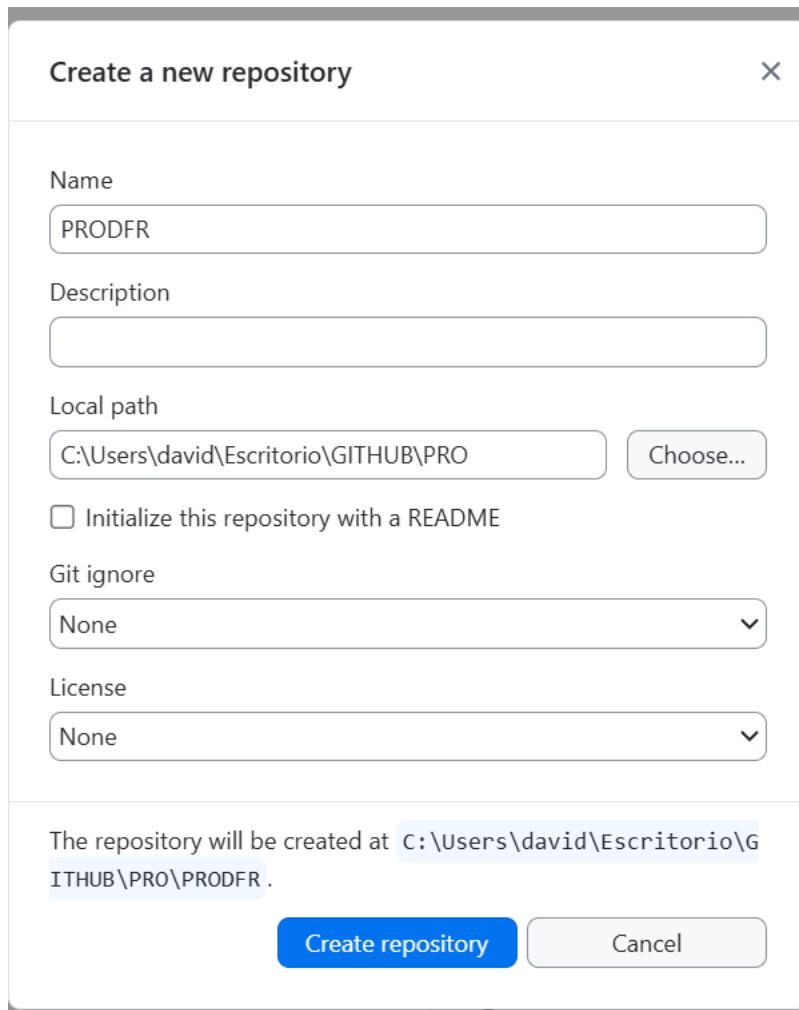


Ilustración 25 Confirmar ubicación y crear repositorio local. Autoría propia

9.-Entramos al panel principal de GitHub y confirmamos que el repositorio PRODFR aparece en la lista junto con nuestros demás proyectos. Esto indica que la creación fue exitosa y ya podemos empezar a trabajar.

The screenshot shows the GitHub homepage. At the top, there's a navigation bar with a menu icon, the GitHub logo, and the word "Panel". Below this, a section titled "Los mejores repositorios" (The best repositories) features a search bar with the placeholder "Encuentra un repositorio..." and a green "Nuevo" button. A list of repositories owned by the user DAVID08FR follows: "proyectoIntegrador", "ProyectoReact", "DavidRepos", "PRODFR", "Reaccionar", "lala", and "sistema". To the right, there's a sidebar titled "Hogar" (Home) with sections for "Pregúntale al copiloto" (Ask the copilot), "Crea un README de perfil para mí" (Create a profile README for me), and "Alimentar" (Feed). Below these are links to "Repositorios de tendencias" (Trending repositories) and a link to "microsoft/rebaiza".

Ilustración 26 Visualización de repositorios en GitHub. Autoría propia

10.-Ubicamos la carpeta donde creamos el repositorio (por ejemplo, PRODFR) y agregamos nuestros archivos del proyecto como index.html, style.css y app.js. Estos archivos se pueden subir y versionar desde GitHub Desktop.

The screenshot shows a Windows File Explorer window. The path is displayed as "PRODFR > GITHUB > PRO > PRODFR". The left sidebar shows icons for Inicio, Galería, Descargas, Escritorio, Música, Imágenes, and Videos. The main area lists files in the "PRODFR" folder: ".gitattributes" (modified 15/07/2025, 11:45 a.m., 1 KB), "# style" (modified 09/07/2025, 02:25 p.m., 2 KB), "app.js" (modified 09/07/2025, 03:09 p.m., 6 KB), and "index" (modified 09/07/2025, 02:25 p.m., 2 KB).

Ilustración 27 Archivos del proyecto dentro del repositorio local. Autoría propia

11.-Revisamos los archivos agregados, escribimos un mensaje en el campo “Summary” (por ejemplo, “Primer commit”) y hacemos clic en “Commit 3 files to main” para guardar los cambios localmente en el repositorio.

```

@@ -0,0 +1,205 @@
+ // Configuración Firebase
+ firebase.initializeApp({
+   apiKey: "AIzaSyCerRjakeR-cfkKhNya4wRCQGQo7z_vZ8",
+   authDomain: "prodfr-3baa2.firebaseio.com",
+   projectId: "prodfr-3baa2",
+   storageBucket: "prodfr-3baa2.appspot.com",
+   messagingSenderId: "1013814917026",
+   appId: "1:1013814917026:web:3d593a802c36ee918124fb",
+   measurementId: "G-2RKL2V44ZC"
+ });
+
+ const auth = firebase.auth();
+ const db = firebase.firestore();
+
+
+ // Bloquear fechas pasadas al cargar la página
+ window.onload = () => {
+   const inputFecha = document.getElementById("fecha");
+   const hoy = new Date().toISOString().split("T")[0];
+   inputFecha.setAttribute("min", hoy);
+ };
+
+ // Validar email
+
+ function esEmailValido(email) {
+   const regex = /^[^@\s]+@[^\s]+\.[^\s@]+\$/;
+   return regex.test(email);
+ }
+
+
+ // Limpiar campos de login/registro
+ function limpiarCamposAuth() {
+   document.getElementById("email").value = "";
+   document.getElementById("password").value = "";
+ }

```

Commit 3 files to main

Committed 3 minutes ago Undo

Initial commit

Ilustración 28 Subir archivos con un commit en GitHub Desktop. Autoría propia

12.-Una vez realizados los commits, GitHub Desktop nos sugiere publicar el repositorio. Presionamos el botón “Publish repository” para que nuestro proyecto esté disponible en GitHub y podamos compartirlo o colaborar con otros.

No local changes

There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.



Publish your repository to GitHub

This repository is currently only available on your local machine. By publishing it on GitHub you can share it, and collaborate with others.

[Publish repository](#)

Always available in the toolbar for local repositories or [Ctrl + P](#)

Open the repository in your external editor

Select your editor in [Options](#)

Repository menu or [Ctrl + Shift + A](#)

[Open in Visual Studio Code](#)

View the files of your repository in Explorer

Repository menu or [Ctrl + Shift + F](#)

[Show in Explorer](#)

Ilustración 29 Opción para publicar el repositorio en GitHub. Autoría propia

13.-En la ventana de publicación, confirmamos el nombre del repositorio (por ejemplo, PRODFR), elegimos si lo queremos público o privado, y presionamos “Publish repository” para subirlo oficialmente a GitHub.

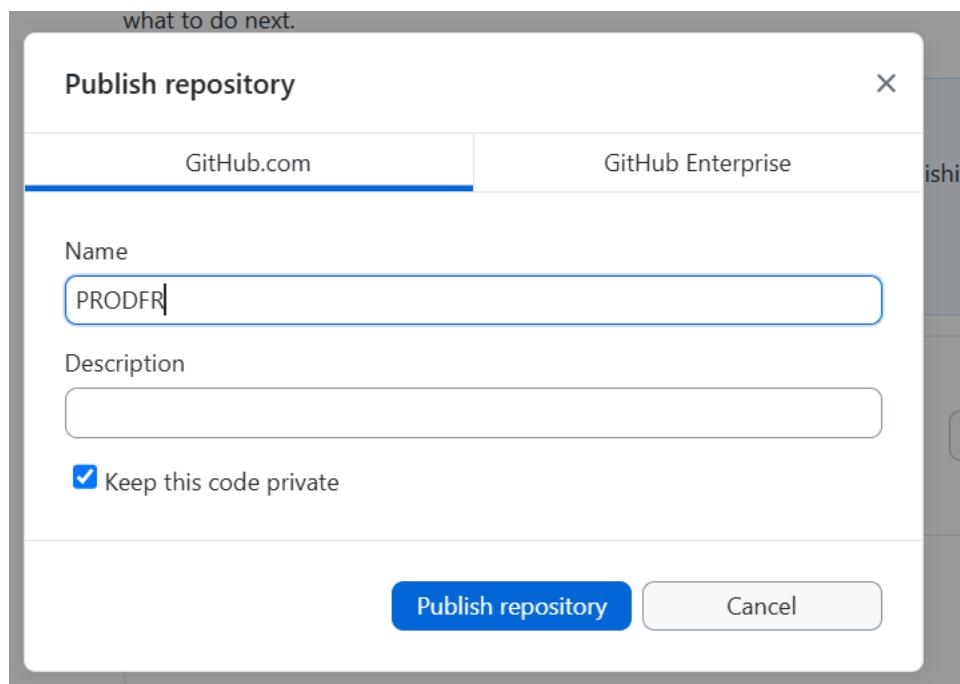


Ilustración 30 Publicar repositorio desde GitHub Desktop. Autoría propia

14.-Entramos a nuestro perfil en GitHub y confirmamos que el repositorio PRODFR aparece publicado, con todos los archivos subidos correctamente y el historial de confirmaciones activo.

A screenshot of a GitHub repository page titled 'PRODFR'. The repository is described as 'Privado'. The main area shows a file list with the following details:

Nombre del archivo	Tipo	Última actualización
DAVID08FR	Primer compromiso	8019c5a · hace 1 minuto
.gitattributes	Compromiso inicial	hace 5 minutos
aplicacion.js	Primer compromiso	hace 1 minuto
índice.html	Primer compromiso	hace 1 minuto
estilo.css	Primer compromiso	hace 1 minuto

On the right side of the page, there are sections for 'Acerca de', 'Lanzamientos', and 'Paquetes'. The 'Acerca de' section notes that no description, website, or topics are provided. The 'Lanzamientos' section indicates there are no published announcements, and the 'Paquetes' section shows no packages.

Ilustración 31 Repositorio publicado en GitHub con archivos cargados. Autoría propia

PROYECTO EN STATIC WEB APPS

Dentro del portal de Azure, escribimos “Static Web Apps” en la barra de búsqueda superior. Seleccionamos el servicio para comenzar la creación de una aplicación web estática con conexión a GitHub.

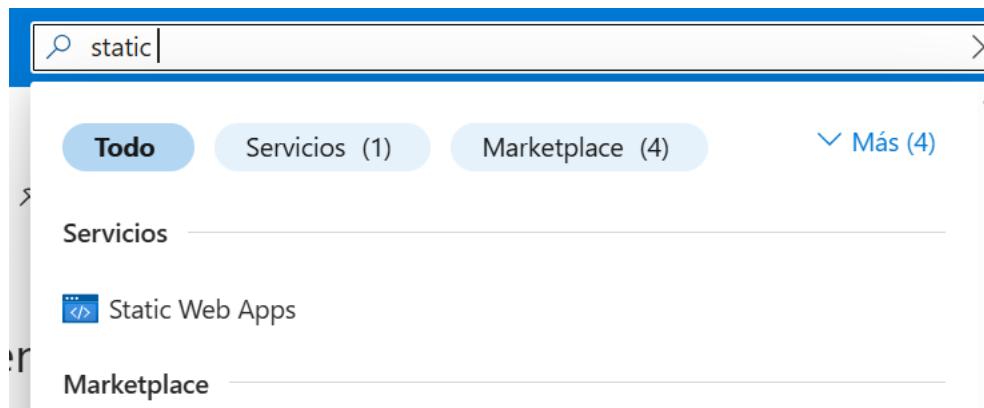


Ilustración 32 Búsqueda del servicio “Static Web Apps” en Azure. Autoría propia

Después de buscar el servicio, accedemos a la sección Static Web Apps en Azure. Si no tenemos aplicaciones aún, veremos este mensaje. Damos clic en el botón “Crear” para iniciar el proceso de publicación de nuestro sitio web.

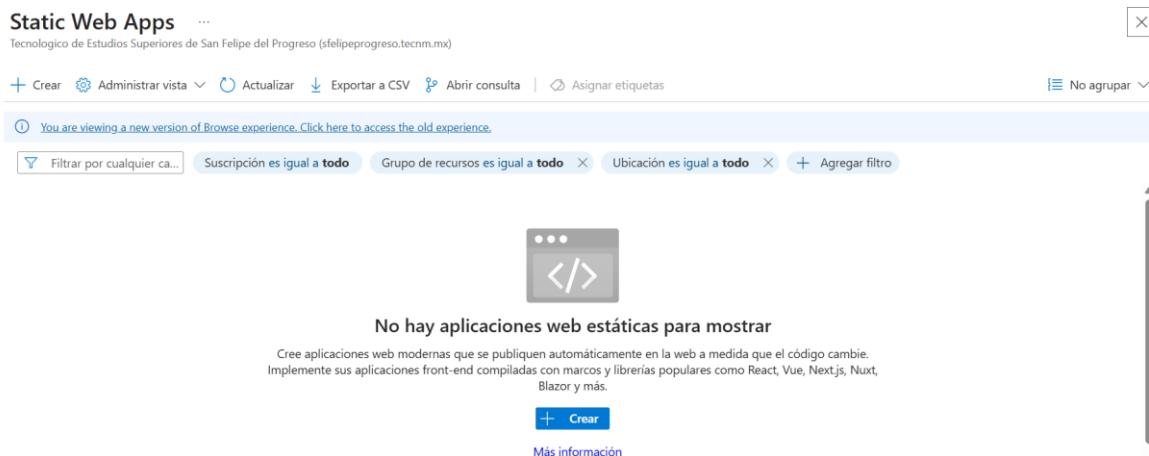


Ilustración 33 Vista inicial de Static Web Apps en Azure. Autoría propia

Llenamos el formulario de creación de la aplicación. Seleccionamos la suscripción Azure for Students, creamos un grupo de recursos, escribimos el nombre del sitio y dejamos el plan gratuito por defecto. Luego damos clic en “Siguiente: Configuración de la implementación”.

Crear una aplicación web estática ...

Suscripción * ⓘ

Azure for Students

Grupo de recursos * ⓘ

(Nuevo) Grupo de recursos

Crear nuevo

Región de hospedaje

Static Web Apps distribuye globalmente los recursos estáticos de la aplicación. Configure las características regionales en [Avanzados](#)

Regiones Global

Detalles de la aplicación web estática

Nombre *

Escriba un nombre para la aplicación web estática

Plan de hospedaje

El plan de hospedaje dicta el ancho de banda, el dominio personalizado, el almacenamiento y otras características disponibles. [Comparar planes](#)

Tipo de plan

Gratis: para aficiones o proyectos personales.

Revisar y crear < Anterior Siguiente: Configuración de la implementación >

Ilustración 34 Formulario para crear una Static Web App. Autoría propia

Cuando se nos pide un grupo de recursos, hacemos clic en “Crear nuevo”, escribimos un nombre (por ejemplo, AzureDFR) y presionamos “Aceptar”. Este grupo será el contenedor donde se almacenarán todos los recursos relacionados con la aplicación.

Región de hospedaje

Static Web Apps distribuye globalmente los recursos en [Avanzados](#)

Regiones

Detalles de la aplicación web estática

Nombre *

Escriba un nombre para la aplicación web estática

Un grupo de recursos es un contenedor que tiene los recursos relacionados de una solución de Azure. regionales

Aceptar Cancelar

Ilustración 35 Creación del grupo de recursos “AzureDFR” . Autoría propia

VINCULAR AZURE – GITHUB

En la sección de origen de código, elegimos GitHub como fuente. Confirmamos nuestra cuenta (por ejemplo, DAVID08FR) y seleccionamos el repositorio PRODFR y la rama main. Esto vincula el proyecto de GitHub con Azure para el despliegue automático.

Origen GitHub Azure DevOps Otro

Cuenta de GitHub DAVID08FR [Cambiar cuenta](#) [?](#)

 Si no encuentra ninguna organización ni repositorio, puede que tenga que habilitar permisos adicionales en GitHub. Debe tener acceso de escritura en el repositorio elegido para realizar la implementación con Acciones de GitHub. [X](#)

Organización *	DAVID08FR
Repository *	PRODF
Rama *	main

Ilustración 36 Selección de repositorio GitHub en Azure. Autoría propia

Establecemos los detalles del flujo de trabajo para compilar y desplegar nuestra app desde GitHub.

Detalles de la compilación

Escriba los valores para crear un archivo de flujo de trabajo de Acciones de GitHub para la compilación y la versión. Puede modificar el archivo de flujo de trabajo más adelante en el repositorio de GitHub.

Valores preestablecidos de compilación

 Estos campos reflejarán la estructura de proyecto predeterminada del tipo de aplicación. Cambie los valores para que se adapten a su aplicación. [Más información](#)

Ubicación de la aplicación *	/
Ubicación de la API	p. ej., "API", "funciones", etc...
Ubicación de salida	.

Configuración del flujo de trabajo

Haga clic en el botón siguiente para obtener una vista previa del aspecto que tendrá el archivo de flujo de trabajo de Acciones de GitHub antes de configurar la implementación continua.

[Vista previa de archivo de flujo de trabajo](#)

[Revisar y crear](#)

[< Anterior](#)

[Siguiente: Configuración de la implementación >](#)

Ilustración 37 Configuración de compilación y flujo de trabajo. Autoría propia

En este paso, Azure realiza una validación automática de todos los datos ingresados. Una vez completada, se habilitará el botón "Crear" para desplegar oficialmente la aplicación estática conectada con GitHub.

Crear una aplicación web estática



Datos básicos Configuración de la implementación Avanzad

Resumen

 **Aplicación web estática**
de Microsoft

Detalles

Ilustración 38 Validación final. Autoría propia

Una vez validada toda la configuración, Azure inicia el proceso de implementación de la plantilla en el grupo de recursos llamado "AzureDFR". Esto indica que tu aplicación estática ya se está construyendo y desplegando automáticamente a partir del repositorio de GitHub.

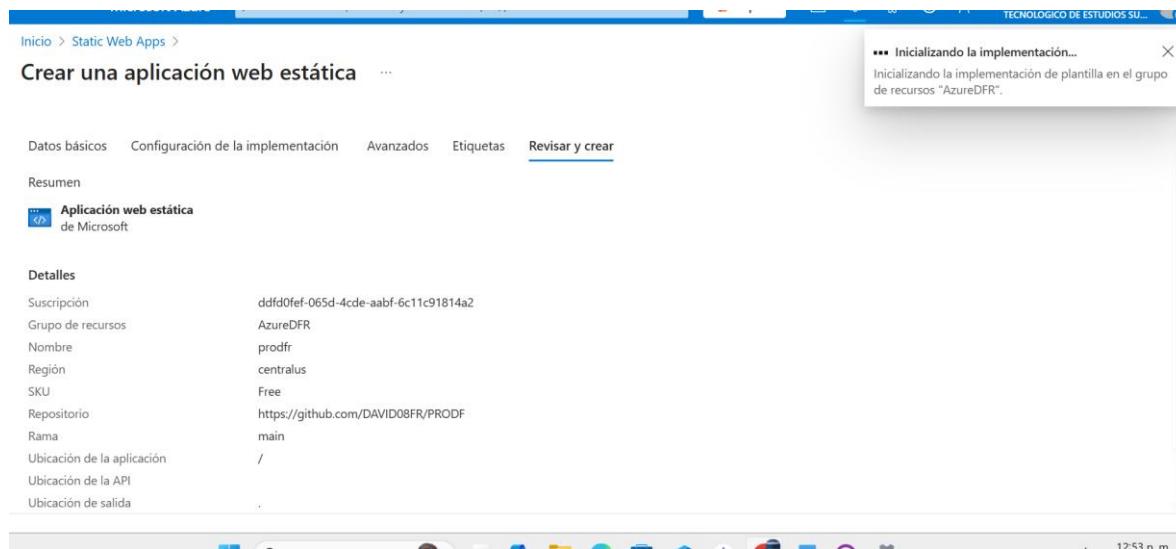


Ilustración 39 Inicialización del despliegue. Autoría propia

La implementación de la aplicación web estática ha finalizado exitosamente. Azure muestra el mensaje “Implementación correcta”. Aquí se confirman los datos clave

The screenshot shows the Microsoft Web Static App Portal interface. At the top, it displays the URL "Microsoft.Web-StaticApp-Portal-a0a9b0e8-9389 | Información general". Below this, a green checkmark indicates "Se completó la implementación". A summary table provides details: Nombre de implementación: Microsoft.Web-Static...; Hora de inicio: 15/7/2025, 12:53:38 p.m.; Suscripción: Azure for Students; Id. de correlación: cc2a38d3-1781-4039-85d4-...; Grupo de recursos: AzureDFR. To the right, a sidebar highlights "Implementación correcta" and "Administración de costos".

Ilustración 40 Implementación correcta. Autoría propia

Una vez finalizada la implementación, se muestra la información esencial del recurso creado. En esta sección se encuentra la dirección URL pública donde se puede ver la aplicación en línea:

<https://salmon-plant-00d705310.2.azurestaticapps.net>

The screenshot shows the Azure portal page for the deployed static app. It includes a toolbar with options like "Ver aplicación en el explorador", "Actualizar", "Eliminar", "Administrar token de implementación", and "Envíenos sus comentarios". The main area displays the "Essentials" section with details: Grupo de recursos (mover) - AzureDFR; Dirección URL - <https://salmon-plant-00d705310.2.azurestaticapps.net>; Suscripción (mover) - Azure for Students; Origen - main (GitHub); Id. de suscripción - ddfd0fef-065d-4cde-aabf-6c11c91814a2; Historial de implementación - Ejecuciones de Acciones de GitHub; Ubicación - Global; Ver flujo de trabajo - azure-static-web-apps-salmon-plant-00d705310.yml; SKU - Free; and Etiquetas (editar) - Agregar etiquetas. Navigation links at the bottom include "Comenzar", "Supervisión", and "Historial de implementación".

Ilustración 41 Ver aplicación en el explorador. Autoría propia

Al abrir la dirección URL proporcionada por Azure, se carga correctamente la aplicación. En este caso, se muestra la pantalla principal del Sistema de Reservas, con una interfaz clara que permite al usuario iniciar sesión o registrarse.



Ilustración 42 Sistema de Reservas - Pantalla de inicio. Autoría propia

Modificamos el código fuente como parte del proceso de implementación continua en Azure Static Web Apps. Este es el tercer commit del proyecto.

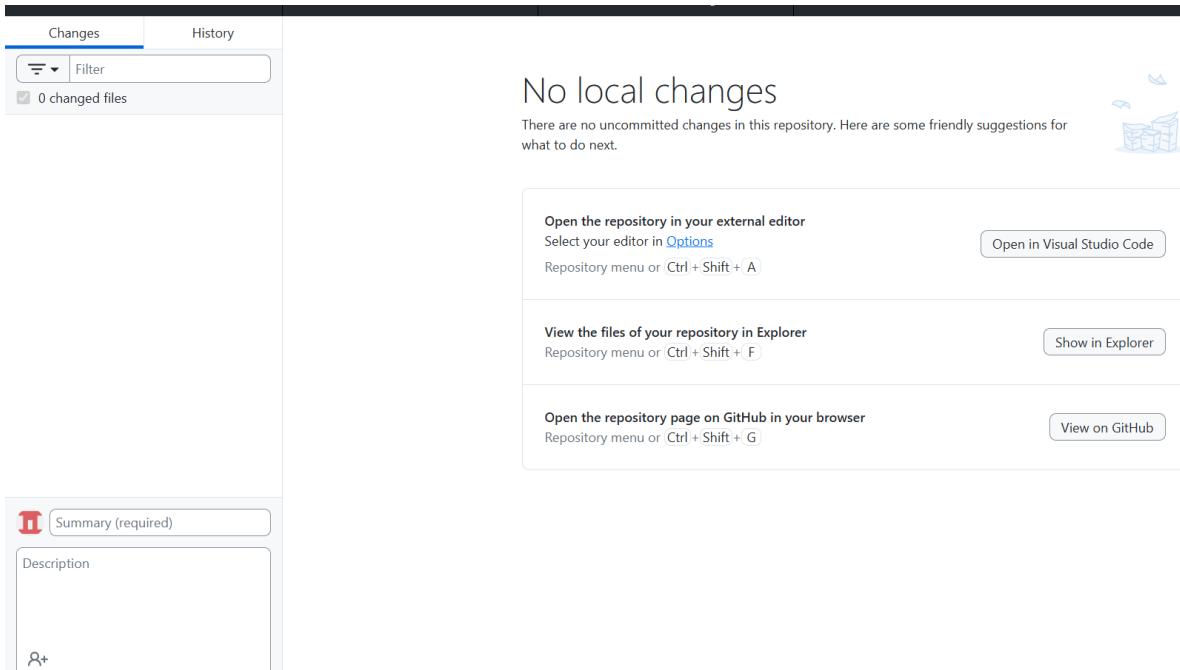


Ilustración 43 cambio en la aplicación. Autoría propia

En la imagen se aprecia que se subió correctamente el tercer commit al repositorio de GitHub con el mensaje "tercer compromiso". Ya aparece reflejado en la rama principal (main), lo que confirma que el cambio fue guardado exitosamente.

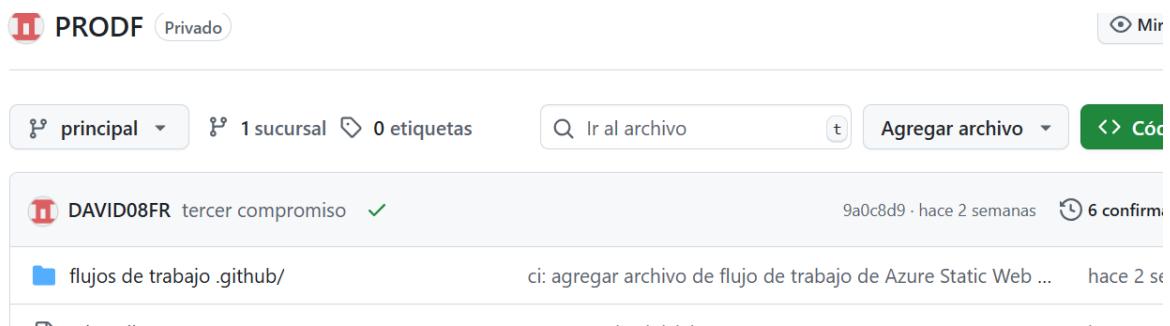


Ilustración 44 Confirmación del tercer compromiso en GitHub. Autoría propia

En la imagen se muestra la sección de detalles de la aplicación web estática desplegada en Azure. Procedemos a presionar el botón "Actualizar" para que los cambios recientes subidos al repositorio se reflejen en la aplicación publicada en el enlace generado (<https://salmon-plant-00d705310.2.azurestaticapps.net>). Esto permite visualizar la versión más reciente del sitio.

Ver aplicación en el explorador Actualizar Eliminar Administrar token de implementación Envíenos sus comentarios

^ Essentials

Grupo de recursos ([mover](#))
[AzureDFR](#)

Suscripción ([mover](#))
[Azure for Students](#)

Id. de suscripción
ddfd0fef-065d-4cde-aabf-6c11c91814a2

Ubicación
Global

SKU
Free

Etiquetas ([editar](#))
[Agregar etiquetas](#)

Dirección URL
<https://salmon-plant-00d705310.2.azurestaticapps.net>

Origen
[main \(GitHub\)](#)

Historial de implementación
[Ejecuciones de Acciones de GitHub](#)

Ver flujo de trabajo
[azure-static-web-apps-salmon-plant-00d705310.yml](#)

Ilustración 45 Actualización de la aplicación web en Azure. Autoría propia

En la imagen se muestra la interfaz actualizada de la aplicación después de los cambios implementados.

Usuario: richi2@gmail.com

dd / mm / aaaa

-- : -- -----

Selecciona un laboratorio ▾ Reservar

Mis Reservas:

2025-07-16 - 16:23 - Lab 1

Cerrar sesión

Ilustración 46 selección de laboratorio. Autoría propia

CREACIÓN DE DNS

Aquí ya entramos al portal de Azure y buscamos “Zona DNS” en la barra de arriba. Aparecen varias opciones como “Zonas DNS” y “Zonas de DNS privado”. Desde aquí elegimos la opción para empezar a crear nuestra zona DNS y conectar el dominio personalizado con nuestra app.

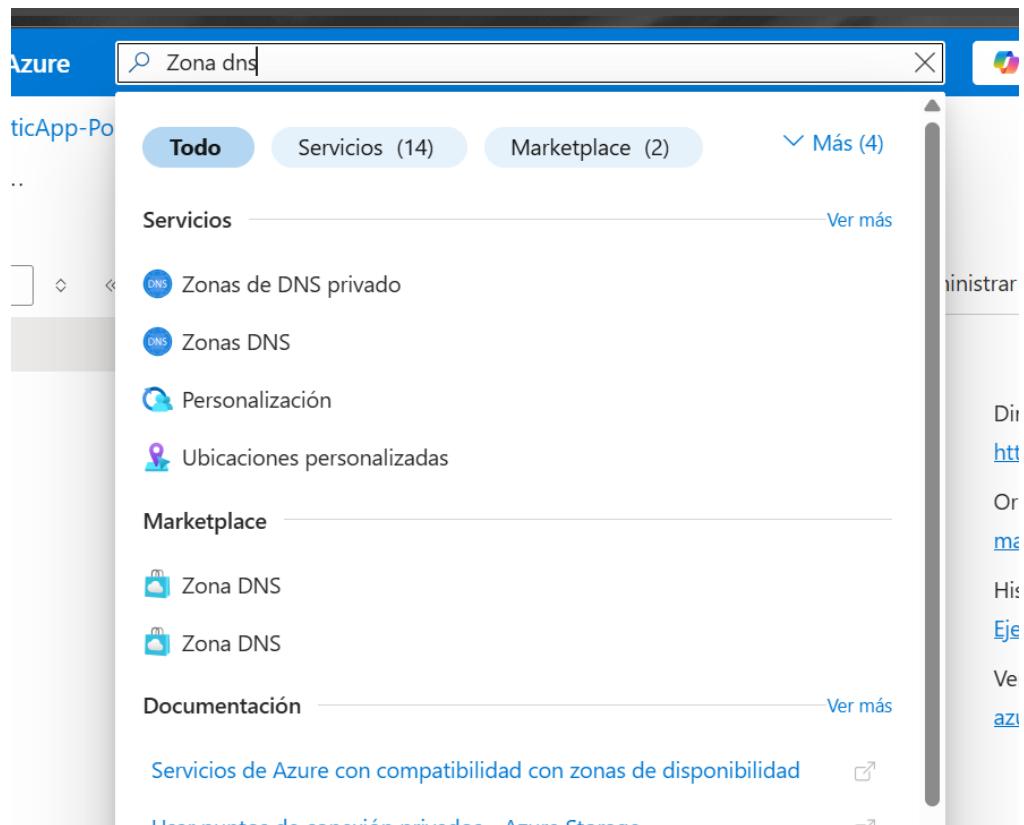


Ilustración 47 Búsqueda de servicio para la creación de Zona DNS en Azure. Autoría propia

Aquí ya entramos al apartado de Hosting en Firebase para crear el dominio de nuestra aplicación. Desde esta sección podemos ver quién hizo el último despliegue, los dominios disponibles y empezar a configurar todo para que nuestra app se publique en línea.

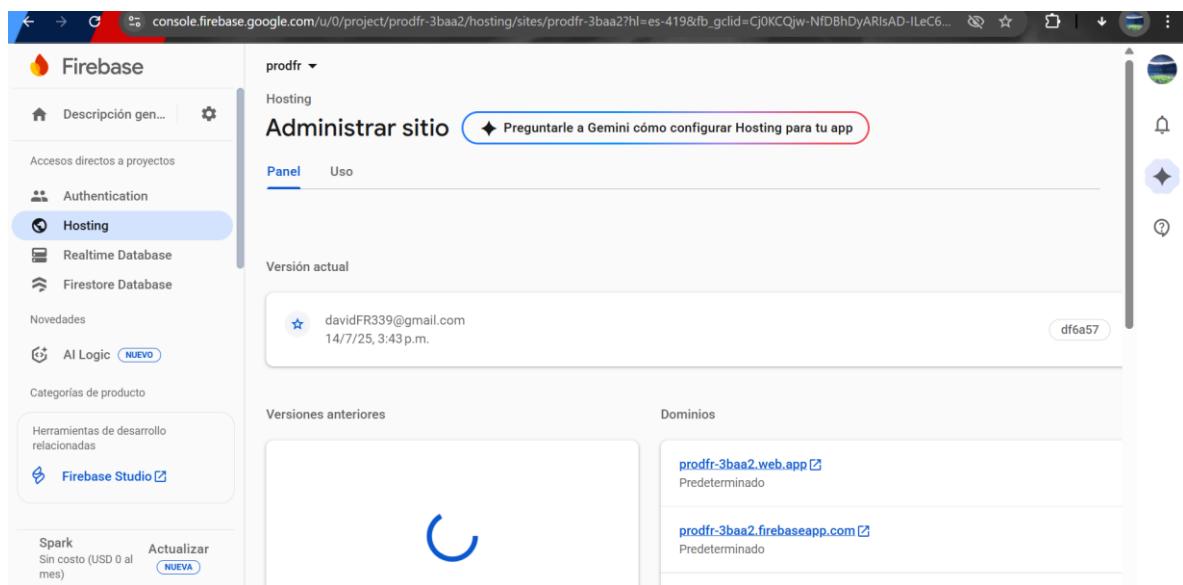


Ilustración 48 Administrando el sitio en Firebase. Autoría propia

Ya dentro del panel de Hosting, le damos clic al botón que dice Agregar un dominio personalizado. Esto nos va a permitir conectar nuestra app con una dirección propia en lugar de usar el dominio por defecto que da Firebase.



Ilustración 49 Agregar un dominio personalizado. Autoría propia

Ahora solo escribimos el dominio que queremos usar, en este caso: reservaciones.com. Luego damos clic en Continuar para seguir con la configuración y conectar el dominio con nuestra app.

Agrega un dominio personalizado

Ingresá el nombre de dominio exacto que usará el público para visitar tu sitio. Puede ser un dominio (tudominio.com) o un subdominio (app.tudominio.com).

Dominio

reservaciones.com

Redirigir **reservaciones.com** a un sitio web existente ?

★ ¿Necesitas un dominio?

Más información ↗

Cancelar

Continuar

Ilustración 50 Escribimos nuestro dominio personalizado. Autoría propia

Después de poner el dominio, Firebase nos da dos registros que tenemos que copiar en nuestro proveedor DNS.

- Uno es un registro A con una dirección IP.
- El otro es un registro TXT que sirve para verificar que somos los dueños del dominio.

Con eso, ya casi está todo listo para que nuestra página funcione con el nuevo dominio personalizado.

Editar dominio reservaciones.com

X

Se agregó el dominio personalizado. Para completar la configuración, realiza los cambios de DNS que se muestran a continuación o vuelve a consultar estas instrucciones de la lista de dominios personalizados en Hosting.

Modo de configuración

Configuración rápida

Avanzado

Agrega los siguientes registros a tu proveedor de DNS para verificar que eres el propietario de **reservaciones.com**

Tipo de registro	Nombre del dominio	Valor
A	reservaciones.com	199.36.158.100 
TXT	reservaciones.com	hosting-site=prodfr-3baa2 

Quita los siguientes registros de tu proveedor de DNS para verificar que eres el propietario de **reservaciones.com**

Tipo de registro	Nombre del dominio	Valor

Ilustración 51 Ya se agregó el dominio, ahora toca configurar el DNS. Autoría propia

Nos vamos a Azure y desde el buscador elegimos Zonas DNS. Luego damos clic en “Crear” y llenamos los campos básicos:

- En la Suscripción, usamos Azure for Students
- En Grupo de recursos, seleccionamos AzureDFR
- Y en Nombre, escribimos nuestro dominio: `reservaciones.com`

Con esto, estamos creando el espacio donde vamos a meter los registros DNS que nos dio Firebase.

Datos básicos Editor de zonas DNS Etiquetas Revisar y crear

organizar y administrar todos los recursos.

Suscripción * ⓘ

Azure for Students

Grupo de recursos * ⓘ

AzureDFR

[Crear nuevo](#)

Detalles de la instancia

Esta zona es un elemento secundario de una zona existente que ya está hospedada en Azure DNS. ⓘ

Nombre *

reservaciones.com

Ubicación del grupo de recursos * ⓘ

(US) Central US

Ilustración 52 zona DNS en Azure. Autoría propia

Ya con todos los datos listos, llegamos al paso final para crear la zona DNS. Azure nos muestra un resumen con la suscripción, el grupo de recursos y el nombre del dominio. Como todo está correcto y la validación fue exitosa, solo queda darle clic al botón Crear.

Validación superada

Datos básicos Editor de zonas DNS Etiquetas Revisar y crear

[Ver plantilla de automatización](#)

Datos básicos

Suscripción	Azure for Students
Grupo de recursos	AzureDFR
Ubicación del grupo de recursos	Central US
Nombre	reservaciones.com

DNS Zone Record Set(s)

Number of record sets	0 record set(s)
-----------------------	-----------------

[Crear](#) < Anterior Siguiente >

Ilustración 53 Confirmamos la zona DNS. Autoría propia

Una vez que le dimos clic a “Crear”, Azure comienza a trabajar en segundo plano. Nos aparece el mensaje “La implementación está en curso”, lo cual quiere decir que se está creando la zona DNS dentro del grupo de recursos que seleccionamos.

Solo hay que esperar unos minutos a que termine este proceso. En cuanto esté lista, ya podremos entrar a la zona para meter los registros que nos dio Firebase.

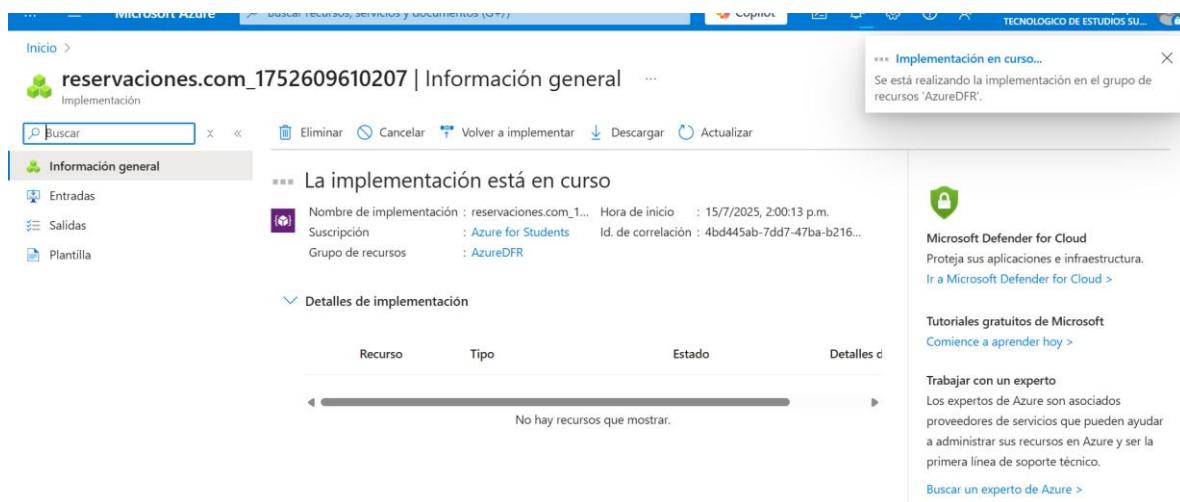


Ilustración 54 Esperamos a que se cree la zona. Autoría propia

Después de unos minutos, Azure nos muestra que la zona `reservaciones.com` ya está lista. Desde aquí podemos ver toda la información importante: los servidores DNS que nos da Azure, cuántos registros hay creados (por ahora solo 2), y que todo quedó dentro del grupo de recursos correcto.

A screenshot of the Microsoft Azure portal showing the 'reservaciones.com' DNS zone. The left sidebar has 'Información general' selected. The main pane shows the 'Essentials' section with the following details:

Configuración	Datos
Zona secundaria	Número máximo de conjuntos de registros: 10000
Conjuntos de registros	Servidor DNS 1: ns1-02.azure-dns.com.
DNSSEC	Servidor DNS 2: ns2-02.azure-dns.net.
Importar	Servidor DNS 3: ns3-02.azure-dns.org.
Exportar	Servidor DNS 4: ns4-02.azure-dns.info.
Mover	
Actualizar	
Eliminar	
Give feedback	

The 'Vis' button is highlighted. At the bottom, there are links for 'Comenzar', 'Tutorial', and 'Herramientas y SDK'.

Ilustración 55 Zona DNS creada con éxito. Autoría propia

VINCULAR AZURE - FIREBASE

Ahora sí, nos vamos a la opción de Registros dentro de la zona DNS. Aquí es donde vamos a meter los datos que nos dio Firebase para que nuestro dominio se conecte correctamente al hosting. Solo hay que dar clic en “+ Agregar conjunto de registros” y poner los dos registros que vimos: el tipo A con la IP 199.36.158.100 y el tipo TXT con el valor que confirma que somos los dueños del dominio.

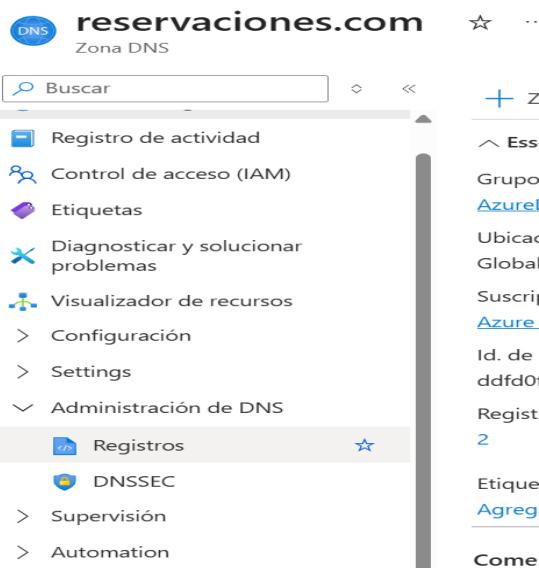


Ilustración 56 agregar los registros DNS. Autoría propia

Entramos al apartado de hosting en Firebase para conectar nuestro dominio personalizado, y luego nos vamos a Azure para crear la zona DNS con ese mismo dominio (reservaciones.com) y agregar los registros que Firebase nos pide (uno tipo A y uno TXT).

The screenshot shows the Firebase DNS management interface. The top navigation bar has 'reservaciones.com' and 'Información general'. The left sidebar includes 'Registro de actividad', 'Control de acceso (IAM)', 'Etiquetas', 'Diagnosticar y solucionar problemas', 'Visualizador de recursos', 'Configuración', 'Settings', 'Administración de DNS', 'Registros' (selected), 'DNSSEC', 'Supervisión', 'Automation', and 'Ayuda'. The main pane shows a search bar and a table of registered records. The table has columns: Nombre, Tipo, TTL, Valor, Tipo de recurso del alias, and Destino c. The records listed are:

Nombre	Tipo	TTL	Valor	Tipo de recurso del alias	Destino c
@	NS	172800	ns1-02.azure-dns.com. ns2-02.azure-dns.net. ns3-02.azure-dns.org. ns4-02.azure-dns.info.		
@	SOA	3600	Email: azuredns-hostmaster.microsoft.com Host: ns1-02.azure-dns.com Refresh: 3600 Retry: 300 Expire: 2419200 Minimum TTL: 300 Serial number: 1		

Ilustración 57 Agregar Registros. Autoría propia

Aquí le metimos el registro A con el nombre www y la IP 199.36.158.100, que es la que nos dio Firebase. Básicamente le estamos diciendo al dominio que apunte directo al servidor donde está nuestra app.

Agregar conjunto de registros

reservaciones.com

Nombre
 .reservaciones.com

Tipo

Conjunto de registros del alias ⓘ

TTL *

Unidad de TTL

Dirección IP
 

Agregar **Cancelar**  Give feedback

Ilustración 58 Agregamos el registro tipo A para apuntar el dominio a Firebase. Autoría propia

En esta imagen se muestra la creación de un nuevo conjunto de registros tipo TXT dentro de la zona DNS del dominio reservaciones.com. Se ha configurado el nombre www y como valor se ingresó la cadena proporcionada por Firebase (hosting-site=pro...). Este registro es necesario para verificar la propiedad del dominio y poder vincularlo correctamente con el servicio de hosting de Firebase.

Agregar conjunto de registros

reservaciones.com

Nombre
.reservaciones.com

Tipo

TTL *

Unidad de TTL

Valor
 [Delete]

El veloz murciélagó hindú comía feliz cardillo y kiwi.

Agregar **Cancelar**  Give feedback

Ilustración 59 Agregamos el registro TXT para validar el dominio en Firebase. Autoría propia

La imagen muestra una herramienta de verificación de propagación DNS, consultando el dominio www.mindsdavidfr.com tipo A. El resultado indica que el dominio aún no se ha propagado correctamente a nivel global, ya que todas las ubicaciones marcadas (Alemania, Sudáfrica, Turquía, Rusia, etc.) muestran fallos de resolución. Esto puede deberse a que el registro DNS aún no se ha actualizado o propagado completamente a los servidores globales.

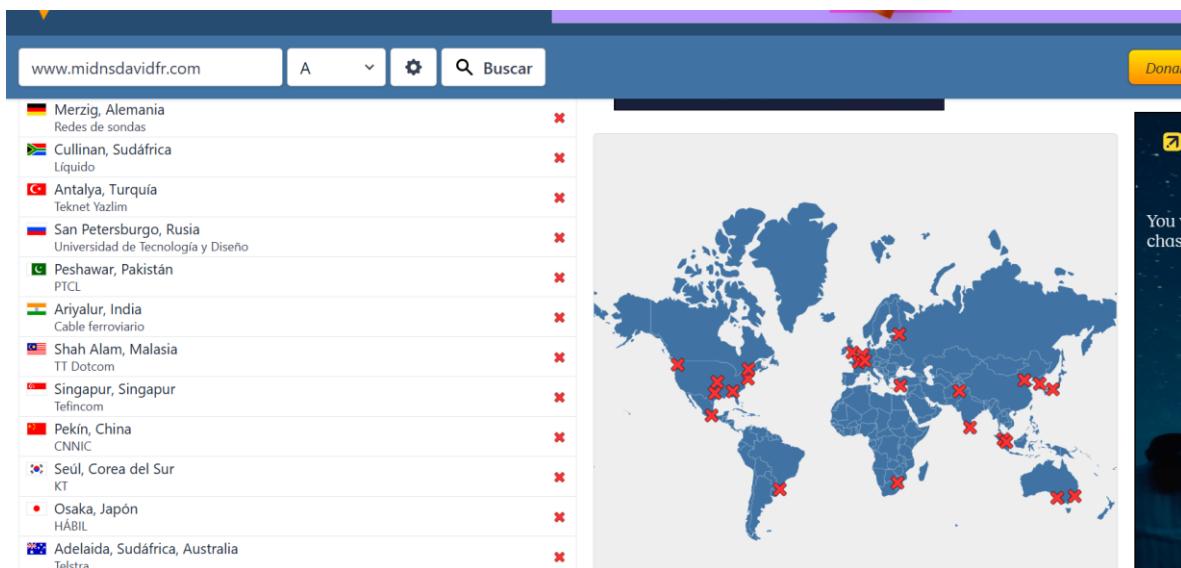


Ilustración 60 Consulta de propagación DNS para el dominio www.mindsdavidfr.com. Autoría propia

Esperar entre 24 y 48 horas tras agregar los registros DNS o verificar que se hayan configurado correctamente en la zona DNS del proveedor (como Azure DNS).

ANÁLISIS DE LA PRÁCTICA

Durante esta práctica se integraron varias tecnologías clave del desarrollo web moderno, permitiendo una experiencia completa y aplicada. Comenzamos con la creación de cuentas en Azure y GitHub, para después desarrollar un proyecto básico que se desplegó en Azure Static Web Apps, conectándolo directamente con el repositorio mediante acciones automatizadas. Esto facilitó que cada cambio hecho en GitHub se viera reflejado automáticamente en el sitio publicado.

Posteriormente, se configuró un dominio personalizado desde el apartado de hosting en Azure, creando la zona DNS y añadiendo los registros necesarios (tipo A y TXT) para validar la propiedad y dirigir correctamente el tráfico. También se hizo una vinculación con Firebase, lo que amplía las posibilidades del proyecto al integrar otros servicios como autenticación o base de datos en tiempo real.

En conjunto, la práctica permitió entender cómo conectar distintos servicios en la nube, automatizar procesos de despliegue, y administrar nombres de dominio. Este tipo de configuración es muy útil para cualquier proyecto web profesional, ya que mejora la eficiencia, escalabilidad y control del desarrollo.

PRACTICA 2

INTRODUCCIÓN

El avance de la computación en la nube ha permitido a estudiantes y profesionales implementar infraestructuras tecnológicas de forma ágil, escalable y económica. Microsoft Azure, como una de las principales plataformas cloud, ofrece servicios esenciales para el desarrollo de soluciones modernas.

En estas prácticas se abordaron dos de los servicios más representativos de Azure: las Máquinas Virtuales (VMs) y el Almacenamiento en Blobs. La primera permite la creación de entornos virtualizados para procesamiento, mientras que la segunda facilita el resguardo y administración de archivos en la nube. Estas herramientas forman parte fundamental de los recursos utilizados en empresas, instituciones educativas y centros de desarrollo.

REQUISITOS GENERALES

Para llevar a cabo ambas prácticas, se requirieron los siguientes elementos:

- Una cuenta activa en **Microsoft Azure** (preferentemente educativa).
- Acceso al **portal de Azure**: <https://portal.azure.com>.
- Conocimientos básicos sobre redes, sistemas operativos y almacenamiento.
- Conexión estable a internet para subir archivos y conectarse a máquinas remotas.

Requisitos específicos por práctica:

- ◆ **Máquina Virtual:**
 - Crear un recurso del tipo VM.
 - Seleccionar sistema operativo (ej. Windows).
 - Configurar autenticación, red, puertos (RDP) y región.
- ◆ **Blob Storage:**
 - Crear una cuenta de almacenamiento.
 - Generar un contenedor tipo **Privado**.
 - Subir un archivo al contenedor (PDF, imagen u otro tipo).

- Consultar propiedades y URL del blob cargado.

CREACION DE MAQUINA VIRTUAL

Paso 1: Generar una clave pública y privada con PuTTYgen

Abrimos la herramienta PuTTY Key Generator (PuTTYgen) para crear un par de claves SSH necesarias para conectarnos de forma segura a la máquina virtual en Azure.

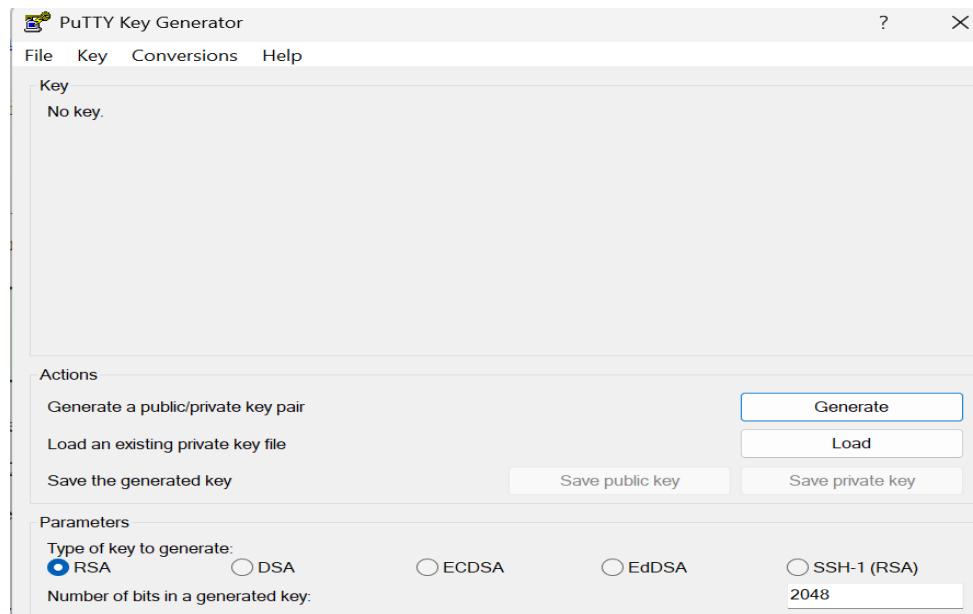


Ilustración 61 Entrar a PUTTYGEN. Autoría propia

Paso 2: Mover el mouse para generar aleatoriedad

Después de hacer clic en Generate, se inicia el proceso de generación de claves. Para completarlo, es necesario mover el cursor del mouse aleatoriamente sobre el área en blanco que aparece en la parte superior

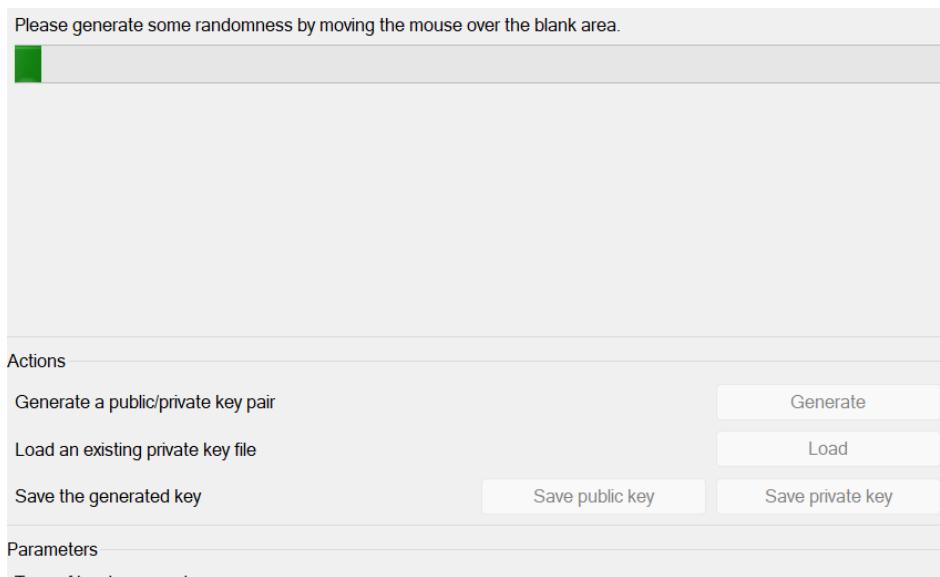


Ilustración 62 Generación de aleatoriedad moviendo el cursor. Autoría propia

Paso 3: Clave SSH generada correctamente

Una vez completada la generación, se muestra la clave pública en la parte superior del programa. Opcionalmente, puedes añadir un comentario en el campo Key comment o proteger la clave privada con una contraseña en Key passphrase.

Es importante hacer clic en Save public key y Save private key para guardar ambos archivos en una ubicación segura, ya que los necesitarás para conectarte a tu máquina virtual en Azure.

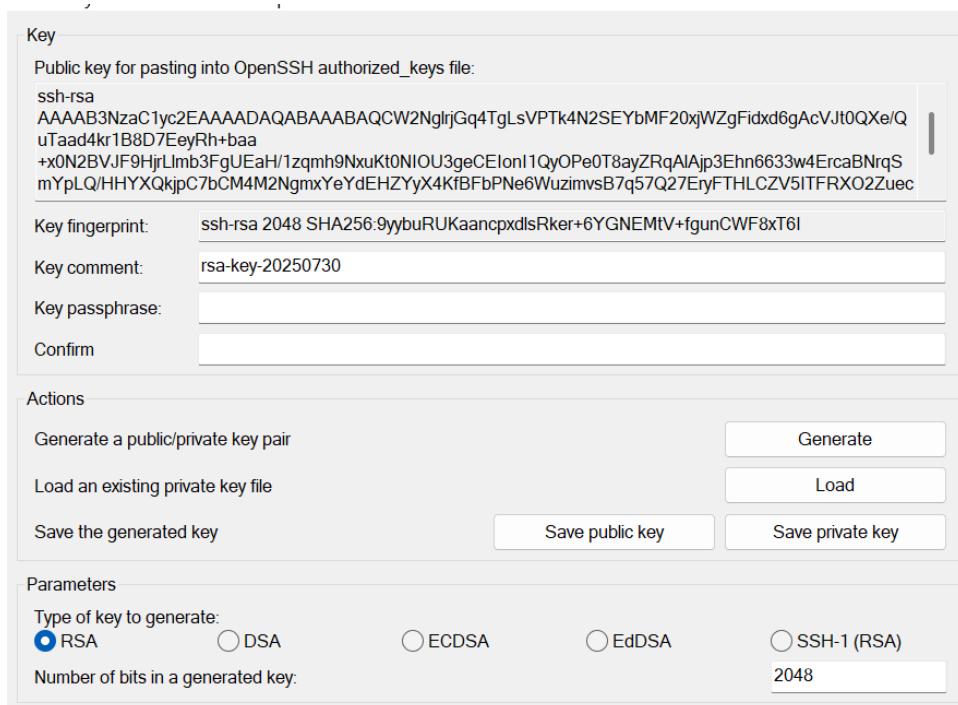


Ilustración 63 Visualización y guardado de la clave pública y privada. Autoría propia

Paso 4: Protección de la clave privada con contraseña

Para proteger la clave privada, puedes ingresar una contraseña (passphrase) en el campo Key passphrase y confirmar en Confirm.

Esto añade una capa adicional de seguridad: cada vez que uses esta clave, se solicitará la contraseña.

Una vez ingresada, guarda la clave pública y privada usando los botones Save public key y Save private key.

The screenshot shows a window for generating a public/private key pair. At the top, there is a large text area containing the key fingerprint: `gC56E0e+A1P4jwylGN0T07jBn83dGRth0AQ04b/0eA/+l6FPDIWbl6E7V0rjA6wB0viODHMMuzPh6dHEG5ovBGSAxAD4oysZ rsa-key-20250730`. Below this, several input fields are displayed:

Key fingerprint:	ssh-rsa 2048 SHA256:9yybuRUKaancpxdlsRker+6YGNEMtV+fgunCWF8xT6I
Key comment:	rsa-key-20250730
Key passphrase:	*****
Confirm:	*****

Below these fields is a section titled "Actions" containing three buttons:

- Generate a public/private key pair (with a "Generate" button)
- Load an existing private key file (with a "Load" button)
- Save the generated key (with "Save public key" and "Save private key" buttons)

At the bottom of the window, there is a "Parameters" section with some dropdown menus.

Ilustración 64 Asignación de passphrase para mayor seguridad. Autoría propia

Paso 5: Guardar la clave privada en tu equipo

Después de generar y proteger tu clave privada, selecciona una ubicación segura en tu equipo para guardarla.

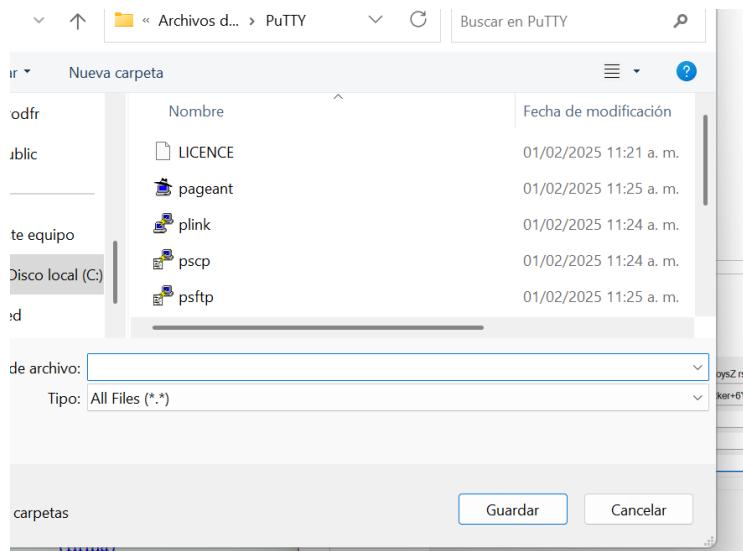


Ilustración 65 Ventana para guardar el archivo de clave privada. Autoría propia

Paso 6: Asignar nombre al archivo de clave privada

En esta pantalla asignamos un nombre al archivo que contiene la clave privada, por ejemplo: prueba123.

Este archivo se guardará con la extensión .ppk, necesaria para conectarnos desde PuTTY a la máquina virtual en Azure.

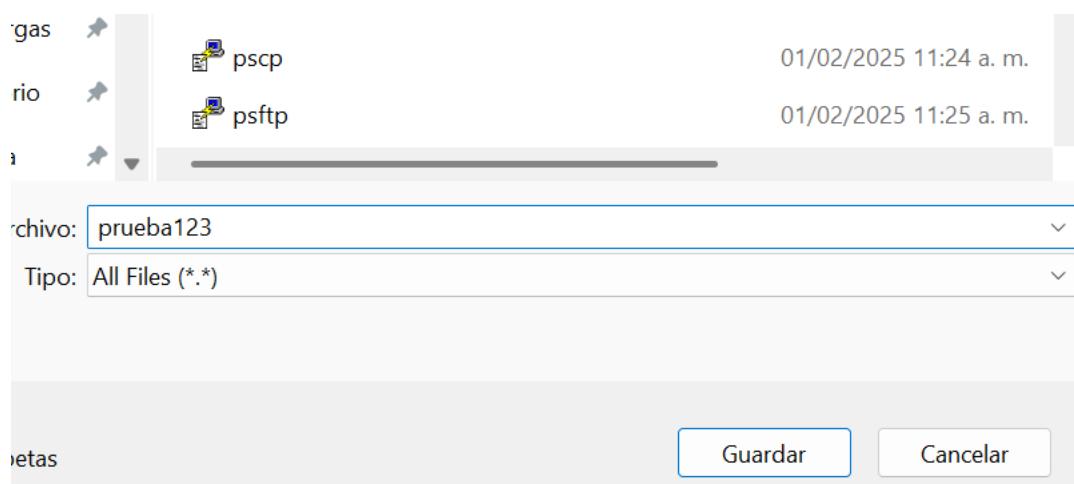


Ilustración 66 Nombrar y guardar el archivo .ppk generado. Autoría propia

Paso 7: Verificar que la clave se guardó correctamente

Una vez guardado el archivo .ppk, podemos verificar que está disponible en el escritorio o en la carpeta que seleccionamos.

En este ejemplo, el archivo fue nombrado como ubuntukey.

Este archivo será utilizado más adelante para acceder a la máquina virtual desde PuTTY. Es importante conservarlo en un lugar seguro.

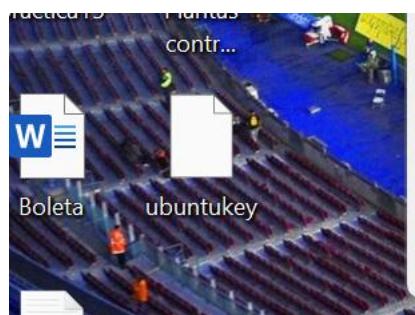


Ilustración 67 Archivo de clave privada almacenado en el escritorio. Autoría propia

Paso 8: Iniciar la creación de una nueva máquina virtual en Azure

Accedemos al portal de Azure desde <https://portal.azure.com>. En la sección “Servicios de Azure”, hacemos clic en la opción Crear un recurso, ubicada al inicio.

Desde aquí comenzaremos la configuración de nuestra nueva máquina virtual que alojará el sistema operativo deseado.

A screenshot of the Azure portal's main dashboard. At the top left, there is a large button with a plus sign and the text 'Crear un recurso'. Below this, there are several service icons: Zonas DNS, Static Web Apps, Centro de inicio rápido, Fundición de IA de Azure, Servicios de Kubernetes, Máquinas virtuales, App Services, Cuentas de almacenamiento, and Más servicios. The 'Crear un recurso' button is highlighted with a white border. The word 'Recursos' is also visible below the main navigation bar.

Ilustración 68 Seleccionar “Crear un recurso” desde el portal de Azure

Paso 9: Seleccionar la opción de creación de máquina virtual

Una vez dentro del portal de Azure, aparecerá una lista de Servicios populares. Localizamos la opción Máquina virtual y hacemos clic en Crear.

Este paso nos llevará al asistente de configuración donde definiremos el sistema operativo, tamaño, autenticación y red de nuestra máquina virtual.

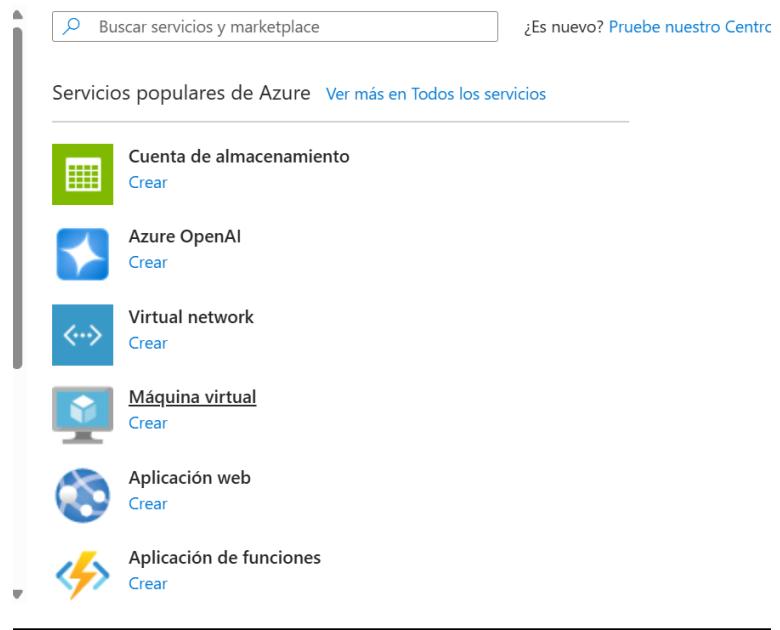


Ilustración 69 Elegir “Máquina virtual” desde servicios populares. Autoría propia

Paso 10: Configurar los datos básicos de la máquina virtual

En esta sección, comenzamos la configuración de la máquina virtual ingresando los datos básicos del proyecto:

- Suscripción: Selecciona la que tengas activa (en este caso, Azure for Students).
- Grupo de recursos: Crea uno nuevo o selecciona uno existente. Este grupo agrupa los recursos relacionados.

[Inicio](#) > [Crear un recurso](#) >

Crear una máquina virtual ...

[Datos básicos](#) [Discos](#) [Redes](#) [Administración](#) [Supervisión](#) [Opciones avanzadas](#) [Etiquetas](#) [Revisar y crear](#)

Cree una máquina virtual que ejecuta Linux o Windows. Seleccione una imagen de Azure Marketplace o use una imagen personalizada propia. Complete la pestaña Conceptos básicos y, después, use Revisar y crear para aprovisionar una máquina virtual con parámetros predeterminados o bien revise cada una de las pestañas para personalizar la configuración.

[Más información ↗](#)

 Es posible que esta suscripción no sea apta para implementar máquinas virtuales de ciertos tamaños en determinadas regiones.

Detalles del proyecto

Seleccione la suscripción para administrar recursos implementados y los costes. Use los grupos de recursos como carpetas para organizar y administrar todos los recursos.

Suscripción *	Azure for Students
Grupo de recursos *	(Nuevo) Grupo de recursos
	Crear nuevo

Detalles de instancia

[< Anterior](#) [Siguiente: Discos >](#) [Revisar y crear](#)

Ilustración 70 Formulario inicial para crear una máquina virtual en Azure. Autoría propia

Paso 11: Configurar detalles de la instancia

- En esta sección definimos las características básicas de la instancia:
- Nombre de máquina virtual: Asigna un nombre identificador, por ejemplo: virtua-Ma.
- Región: Selecciona la región geográfica donde se desplegará la máquina (en este caso, (US) West US 3).
- Zona de disponibilidad: Elige una zona para mejorar la tolerancia a fallos, por ejemplo: Zona 1.
- Tipo de seguridad: Se puede dejar la opción predeterminada “Máquinas virtuales de inicio seguro”.

Detalles de instancia

Nombre de máquina virtual *	<input type="text" value="virtua-Ma"/>
Región *	<input type="text" value="(US) West US 3"/>
Opciones de disponibilidad	<input type="text" value="Zona de disponibilidad"/>
Opciones de zona	<input checked="" type="radio"/> Zona autoseleccionada Elija hasta 3 zonas de disponibilidad, una máquina virtual por zona
	<input type="radio"/> Zona seleccionada por Azure (versión preliminar) Permitir que Azure asigne la mejor zona para sus necesidades
Zona de disponibilidad *	<input type="text" value="Zona 1"/>
	Ahora puede seleccionar varias zonas. Si selecciona varias zonas, se creará una VM por zona. Más información
Tipo de seguridad	<input type="text" value="Máquinas virtuales de inicio seguro"/> Configurar características de seguridad

Ilustración 71 Asignar nombre, región y zona de disponibilidad a la VM

Paso 12: Seleccionar la imagen del sistema operativo y arquitectura

En esta sección seleccionamos la imagen del sistema operativo que se instalará en la máquina virtual:

Imagen: Elegimos Ubuntu Server 24.04 LTS - x64 gen. 2, una de las versiones más estables y modernas para entornos de desarrollo y servidores.

Opciones de disponibilidad	<input type="text" value="Zona de disponibilidad"/>
Opciones de zona	<input checked="" type="radio"/> Zona autoseleccionada Elija hasta 3 zonas de disponibilidad, una máquina virtual por zona
Zona de disponibilidad *	<input type="text" value="Zona 1"/>
	Ahora puede seleccionar varias zonas. Si selecciona varias zonas, se creará una VM por zona. Más información
Tipo de seguridad	<input type="text" value="Máquinas virtuales de inicio seguro"/> Configurar características de seguridad
Imagen *	<input type="text" value="Ubuntu Server 24.04 LTS - x64 gen. 2"/> Ver todas las imágenes Configurar la generación de máquinas virtuales
Arquitectura de VM	<input type="radio"/> Arm64 <input checked="" type="radio"/> x64

Ilustración 72 Elegir Ubuntu como sistema operativo para la VM. Autoría propia

Paso 13: Configurar la cuenta de administrador y autenticación SSH

En esta sección configuramos cómo se accederá a la máquina virtual. Se elige Clave pública SSH como método de autenticación, se asigna un nombre de usuario y se define el tipo de clave SSH que se usará para conectarse.

También se puede generar un par de claves automáticamente o cargar una existente previamente creada con PuTTYgen.

Cuenta de administrador

Tipo de autenticación (i)

Clave pública SSH
 Contraseña

i Ahora, Azure genera automáticamente un par de claves SSH y le permite almacenarlo para usarlo en el futuro. Es una forma rápida, sencilla y segura de conectarse a la máquina virtual.

Nombre de usuario * (i)

daviduser ✓

Origen de clave pública SSH

Generar un par de claves nuevo (i)

Tipo de clave SSH

Formato RSA SSH
 Formato Ed25519 SSH

i Ed25519 proporciona un nivel de seguridad fijo de no más de 128 bits para claves de 256 bits, mientras que RSA podría ofrecer una mejor seguridad con claves de más de 3072 bits.

Nombre de par de claves *

virtua-Ma_key ✓

Ilustración 73 Asignar usuario y método de autenticación a la VM. Autoría propia

Paso 14: Ingresar una clave pública SSH existente

Si ya generaste una clave pública SSH con herramientas como PuTTYgen, puedes pegarla directamente en este campo. Solo selecciona la opción Usar la clave pública existente y copia el contenido completo que generaste. Esta clave permitirá conectarte a tu máquina virtual desde tu equipo de forma segura, sin necesidad de contraseña.

Origen de clave pública SSH

Usar la clave pública existente (i)

i Los formatos Ed25519 y RSA SSH son compatibles con la imagen de VM seleccionada. Ed25519 proporciona un nivel de seguridad fijo de no más de 128 bits para claves de 256 bits, mientras que RSA podría ofrecer una mejor seguridad con claves de más de 3072 bits.

Clave pública SSH * (i)

ssh-rsa
AAAAAB3NzaC1yc2EAAAQABAAQCW2NglrjGq4TgLsVPTk4N2SEYbMF
20xjWZgFidxd6gAcVjt0QXe/QuTaad4kr1B8D7EeyRh+baa+x0N2BVJF9HjrLlmb

i Más información sobre la creación y el uso de claves SSH en Azure ↗

Ilustración 74 Paso 14: Ingresar una clave pública SSH existente. Autoría propia

Paso 15: Configurar los discos de la máquina virtual

En la pestaña Discos, configuramos el tipo de almacenamiento que usará la máquina virtual:

Cifrado del disco: Azure cifra automáticamente los discos administrados en la nube.

Tamaño del disco del SO: Puedes dejar el valor predeterminado de la imagen (30 GiB).

Tipo de disco: Se recomienda usar SSD Premium para un mejor rendimiento, aunque también se puede seleccionar HDD o SSD estándar, según el presupuesto y necesidades.

The screenshot shows the 'Disks' tab selected in the Azure portal. It includes sections for 'Cifrado del disco de la máquina virtual' (which is disabled), 'Disco del SO' (operating system disk) settings (size 30 GiB, type SSD Premium), and a 'Revisar y crear' (Review + Create) button.

Ilustración 75 Seleccionar opciones de almacenamiento para el sistema operativo. Autoría propia

Paso 16: Elegir el tipo de disco del sistema operativo

Azure permite elegir entre distintos tipos de disco para el sistema operativo de la máquina virtual. Las opciones disponibles incluyen:

- SSD Premium: Ideal para cargas de trabajo exigentes y entornos de producción.
- SSD estándar: Recomendado para pruebas, desarrollo y uso moderado.
- HDD estándar: Adecuado para almacenamiento no crítico o de bajo costo.

Almacenamiento con redundancia de zona: Ofrece mayor disponibilidad al replicar datos en múltiples zonas.

Para un buen rendimiento general, se recomienda SSD Premium (almacenamiento con redundancia local).

Disco del SO

Tamaño del disco del SO ⓘ Valor predeterminado de la imagen (30 GiB) ▾

Tipo de disco del sistema operativo * ⓘ SSD Premium (almacenamiento con redundancia local) ▾

Eliminar con VM ⓘ Almacenamiento con redundancia local (los datos se replican en un único centro de datos).

Administración de claves ⓘ SSD Premium
Recomendado para las cargas de trabajo que requieren un alto rendimiento y de producción

Habilitar compatibilidad con Ultra Disks ⓘ SSD estándar
Recomendado para servidores web, aplicaciones empresariales que se usan poco y el desarrollo y las pruebas

Discos de datos para virtua-Ma

Puede agregar y configurar discos de datos tambien incluye un disco temporal.

LUN	Nombre	Tar

[Crear y adjuntar un nuevo disco](#) Asociar

[< Anterior](#) [Siguiente: Redes >](#) [Revisar y crear](#) ▾

Ilustración 76 Seleccionar tipo de almacenamiento para el sistema operativo. Autoría propia

Paso 17: Configurar la interfaz de red de la máquina virtual

Azure configura automáticamente la red para la máquina virtual, pero también puedes personalizarla.

Interfaz de red

Al crear una máquina virtual, se crea una interfaz de red automáticamente.

Red virtual * ⓘ (nuevo) virtua-Ma-vnet ▾
[Crear nuevo](#)

Subred * ⓘ (nuevo) default (10.0.0.0/24) ▾

IP pública ⓘ (nuevo) virtua-Ma-ip ▾
[Crear nuevo](#)

Grupo de seguridad de red de NIC ⓘ Ninguno Básico Opciones avanzadas

Puertos de entrada públicos * ⓘ Ninguno Permitir los puertos seleccionados

Selección de puertos de entrada * ⓘ 554 / 721 ▾

Ilustración 77 Definir red virtual, subred e IP pública. Autoría propia

Paso 18: Revisar configuración y crear la máquina virtual

En la pestaña Revisar y crear, Azure muestra un resumen de toda la configuración seleccionada: sistema operativo, red, discos, autenticación, seguridad, tamaño de la VM y precio estimado por hora. Si todo es correcto y aparece la leyenda Validación superada, haz clic en el botón Crear para iniciar la implementación de la máquina virtual.

Crear una máquina virtual ...

✓ Validez superada

Datos básicos Discos Redes Administración Supervisión Opciones avanzadas Etiquetas **Revisar y crear**

Precio

1 X Standard D2s v3 por Microsoft Se aplican créditos de suscripción ⓘ 0,1110USD/h Términos de uso | Directiva de privacidad Precios de otros tamaños de máquinas virtuales

TÉRMINOS

Al hacer clic en "Crear", (a) acepto los términos legales y las declaraciones de privacidad relacionados con cada oferta de Marketplace que se enumeró previamente; (b) autorizo a Microsoft a facturar con mi método de pago actual las cuotas relacionadas con las ofertas, con la misma frecuencia de facturación que mi suscripción de Azure; y (c) autorizo a Microsoft a compartir mi información de contacto y los datos de transacción y uso con los proveedores de dichas ofertas. Microsoft no proporciona derechos sobre ofertas de terceros. Para obtener información adicional, consulte los [Términos de Azure Marketplace](#).

< Anterior Siguiente > **Crear** Descargar una plantilla para

Paso 19: Confirmar contacto y advertencia de seguridad. Autoría propia

Antes de crear la máquina virtual, Azure solicita confirmar los datos de contacto, como el nombre, correo electrónico y número de teléfono.

Además, se muestra una advertencia indicando que el puerto SSH (22) estará abierto a internet. Esto es necesario para conectarse remotamente, pero se recomienda limitarlo solo a pruebas o restringir el acceso desde una IP específica más adelante.

Una vez verificada esta información, se puede hacer clic en Crear para lanzar la máquina virtual.

Crear una máquina virtual

Validación superada

compartir mi información de contacto y los datos de transacción y uso con los proveedores de algunas ofertas. Microsoft no proporciona derechos sobre ofertas de terceros. Para obtener información adicional, consulte los [Términos de Azure Marketplace](#).

Nombre	David Felipe Roman
Dirección de correo electrónico preferida	i2021330135@sfelipeprogreso.tecnm.mx
Número de teléfono preferido	7121241030

⚠ Ha establecido los siguientes puertos abiertos para Internet: SSH. Esto solo se recomienda para las pruebas. Si quiere cambiar esta configuración, vuelva a la pestaña de aspectos básicos.

Datos básicos

Suscripción	Azure for Students
-------------	--------------------

< Anterior Siguiente > **Crear**

Ilustración 78 Verificación de información y advertencia sobre puertos abiertos. Autoría propia

Paso 20: Verificar el despliegue de la máquina virtual

Después de hacer clic en Crear, Azure inicia el proceso de despliegue de la máquina virtual. En esta pantalla puedes ver el progreso de la implementación, el grupo de recursos utilizado, la IP pública, la red virtual y el grupo de seguridad. Cada recurso aparecerá como Created una vez finalizado correctamente.

Una vez completado este proceso, la máquina virtual estará lista para ser utilizada.

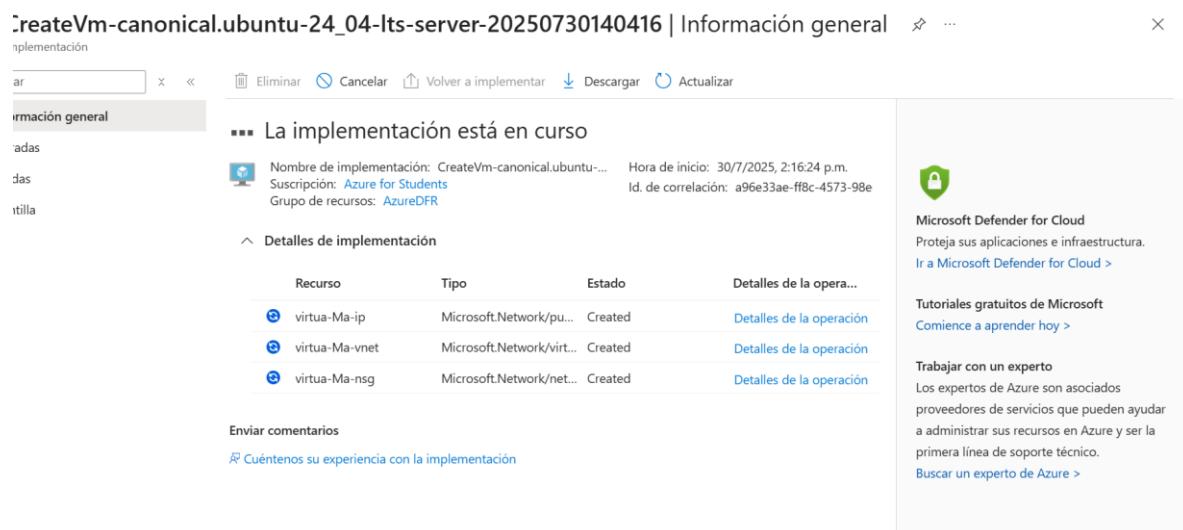


Ilustración 79 La implementación de la máquina virtual está en curso. Autoría propia

Paso 21: Confirmación de implementación exitosa

Una vez finalizado el proceso, Azure muestra una notificación indicando que la máquina virtual se creó con éxito. Desde esta pantalla puedes acceder al recurso, revisar su estado o configurar funciones adicionales como el apagado automático.

Esto indica que la máquina está lista para usarse.

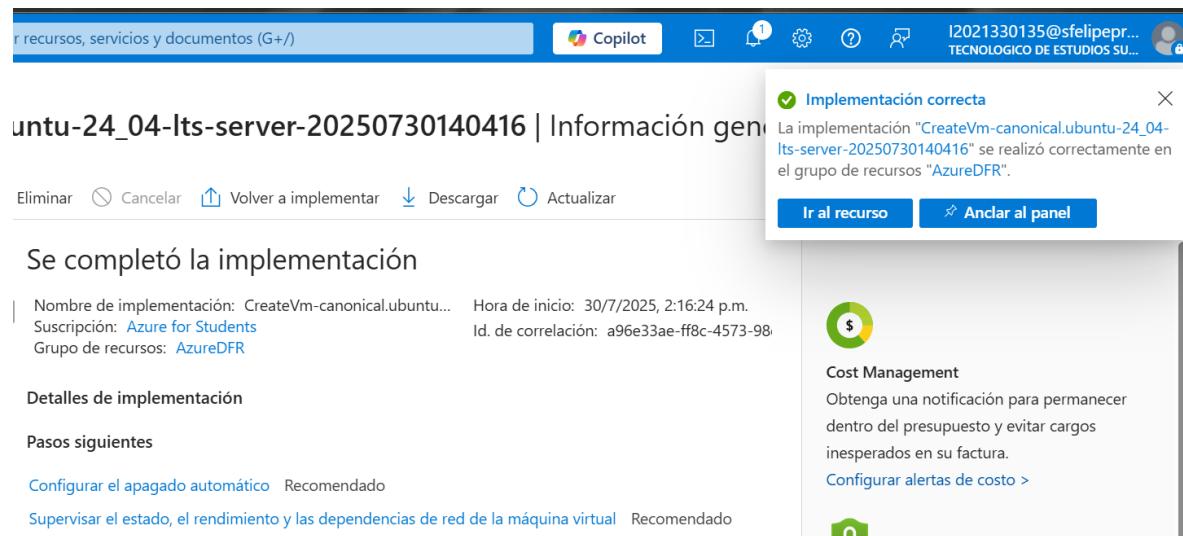


Ilustración 80 Se completó correctamente la creación de la máquina virtual. Autoría propia

Paso 22: Ver detalles de la máquina virtual en ejecución

En esta pantalla puedes ver que la máquina virtual se creó correctamente y ya está activa. Aparecen datos como el sistema operativo, la IP pública, el estado, la ubicación y otros detalles básicos.

Desde aquí puedes conectarte o hacer cambios si es necesario.

Conectar ▾ ▶ Iniciar ⏪ Reiniciar ⏸ Detener ⏹ Hibernar ⏷ Captura ▾ ⏴ Eliminar ⏵ Actualizar ⏵ Abrir en dispositivos m

^ Información esencial Vista

Grupo de recursos (mover)	Sistema operativo
AzureDFR	Linux (ubuntu 24.04)
Estado	Tamaño
En ejecución	Standard D2s v3 (2 vcpu, 8 GiB de memoria)
Ubicación	Dirección IP pública
Canada Central (Zona 1)	20.151.72.3
Suscripción (mover)	Red virtual/subred
Azure for Students	virtua-Ma-vnet/default
Id. de suscripción	Nombre DNS
ddfd0fef-065d-4cde-aabf-6c11c91814a2	Sin configurar
Zona de disponibilidad	Estado de mantenimiento
1	-
Etiquetas (editar)	Hora de creación
Agregar etiquetas	30/7/2025, 8:16 p.m. UTC

Ilustración 81 Información general de la máquina virtual activa. Autoría propia

Paso 23: Cargar la clave privada en PuTTY

En esta parte de PuTTY, se carga la clave privada generada anteriormente para poder conectarse a la máquina virtual. Solo hay que hacer clic en Browse..., buscar el archivo .ppk y luego continuar con la conexión.

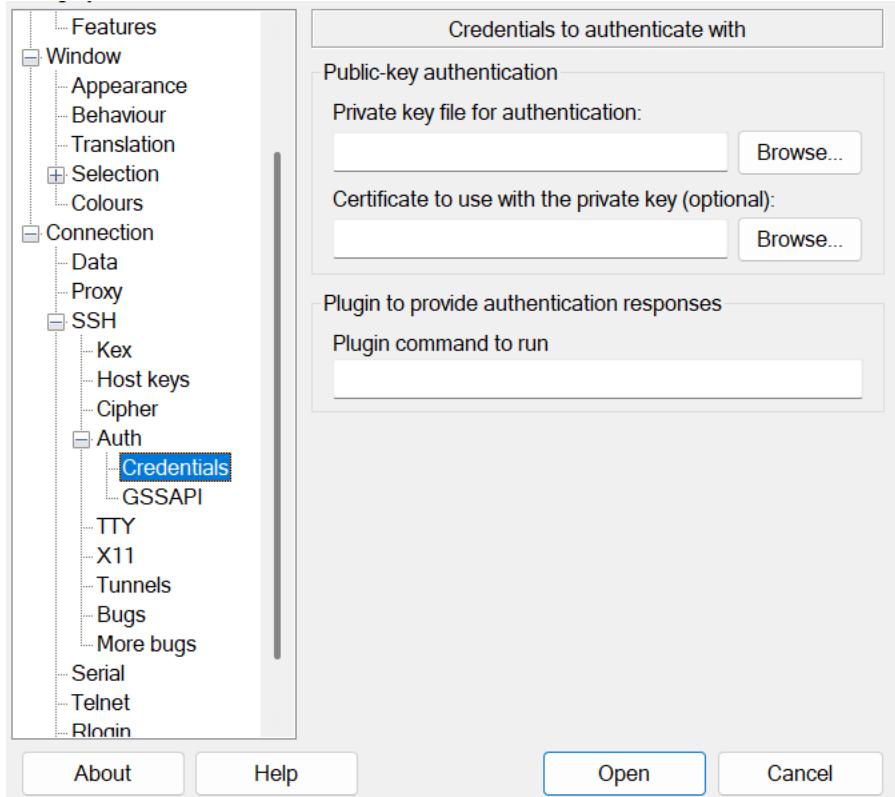


Ilustración 82 Seleccionar archivo de clave privada para la conexión. Autoría propia

Paso 24: Confirmar que la clave privada está cargada en PuTTY

Aquí se muestra que el archivo de clave privada (ubuntukey.ppk) ya ha sido cargado correctamente en PuTTY. Esto permitirá conectarse de forma segura a la máquina virtual en Azure sin usar contraseña.

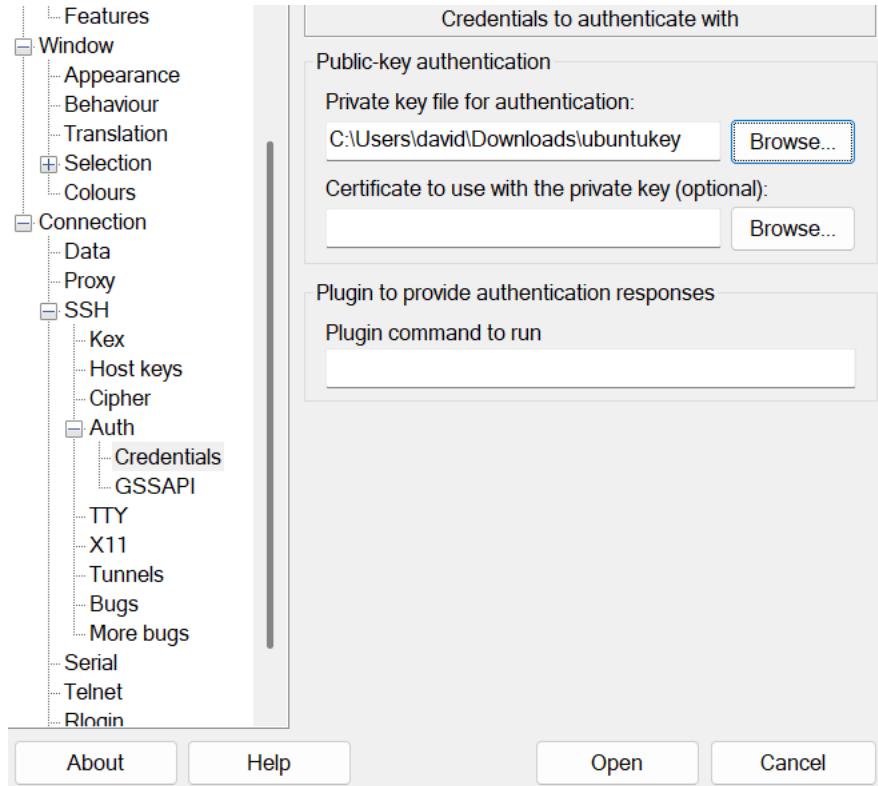


Ilustración 83 Clave privada seleccionada para autenticación SSH. Autoría propia

Paso 25: Especificar el nombre de usuario en PuTTY

Antes de conectar, se debe escribir el **nombre de usuario configurado en Azure** (en este caso, **daviduser**) en el campo **Auto-login username**. Esto le indica a PuTTY con qué usuario iniciar sesión en la máquina virtual.

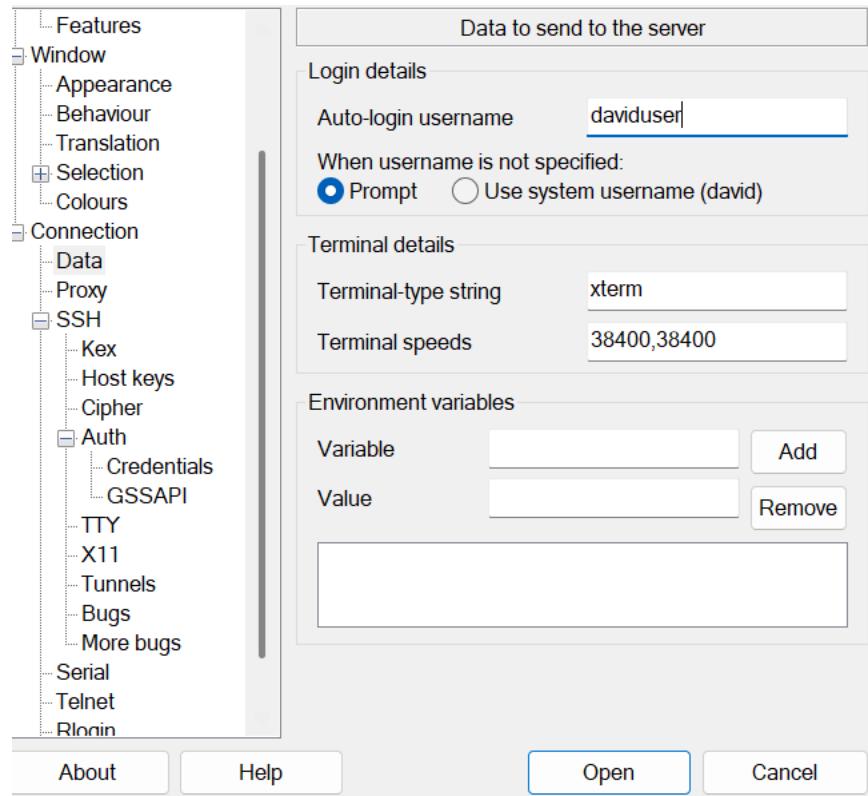


Ilustración 84 Ingresar el nombre de usuario para la conexión SSH. Autoría propia

Paso 26: Conectarse a la máquina virtual desde PuTTY

En esta ventana de PuTTY se configura la conexión final. Se debe ingresar la **IP pública de la máquina virtual** (por ejemplo, 20.151.72.3), asegurarse de que el **puerto sea 22** y el **tipo de conexión SSH** esté seleccionado. También puedes escribir un nombre en “Saved Sessions” y hacer clic en **Save** para guardar la sesión y reutilizarla después fácilmente.

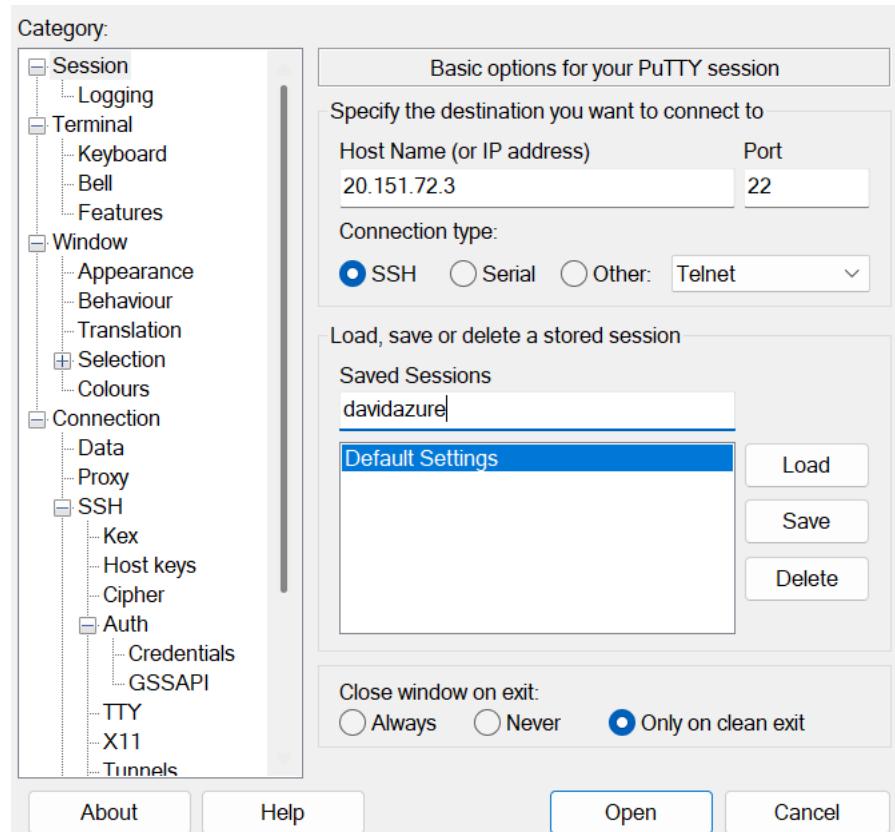


Ilustración 85 Ingresar IP pública y guardar sesión SSH. Autoría propia

Paso 27: Guardar la sesión SSH para futuras conexiones

Una vez configurada la IP y el tipo de conexión, puedes guardar la sesión con un nombre (por ejemplo, davidazure). Esto permite reutilizar todos los datos sin tener que configurarlos cada vez. Solo selecciona la sesión y haz clic en **Save**.

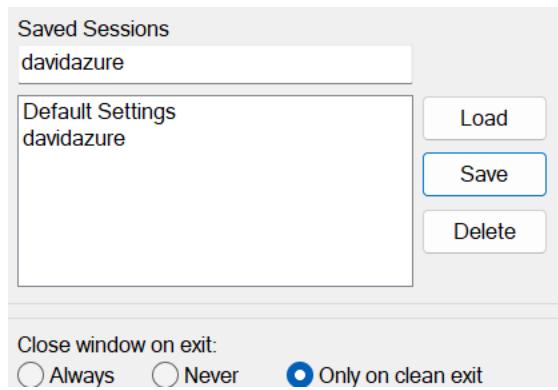


Ilustración 86 Sesión SSH guardada con nombre personalizado. Autoría propia

Paso 28: Ingresar la passphrase para acceder a la máquina virtual

Al hacer clic en Open, se abre la terminal y se inicia la conexión a la máquina virtual usando el nombre de usuario y la clave SSH.

Si configuraste una passphrase al generar la clave privada, se te pedirá que la ingreses para completar la autenticación.

Después de introducirla correctamente, tendrás acceso completo a tu máquina virtual

```
[?] Using username "felipeuser".  
[?] Authenticating with public key "rsa-key-20250730"  
[?] Passphrase for key "rsa-key-20250730": [REDACTED]
```

Ilustración 87 Conexión establecida: autenticación mediante clave SSH. Autoría propia

Paso 29: ¡Conexión exitosa a la máquina virtual!

Después de ingresar la passphrase correctamente, accediste a la máquina virtual. Esta pantalla muestra que el sistema está activo y listo para ejecutar comandos.

Desde aquí puedes actualizar paquetes, instalar software o administrar tu servidor Linux de forma remota.

```
Expanded Security Maintenance for Applications is not enabled.  
0 updates can be applied immediately.  
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status  
  
The list of available updates is more than a week old.  
To check for new updates run: sudo apt update  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
felipeuser@virtua-Machine:~$
```

Ilustración 88 Sesión iniciada en Ubuntu desde PuTTY. Autoría propia

Paso 30: Ejecutar comandos dentro de la máquina virtual

Una vez dentro de la máquina virtual, puedes ejecutar comandos como sudo apt update y sudo apt upgrade para mantener el sistema actualizado.

En esta imagen se muestra cómo el sistema realiza configuraciones y reinicia servicios importantes.

Desde aquí puedes administrar completamente tu servidor Ubuntu.

```

/system-generators/netplan.usr-is-merged by netplan-generator'
Setting up systemd-resolved (255.4-1ubuntu8.10) ...
Setting up systemd-sysv (255.4-1ubuntu8.10) ...
Setting up libnss-systemd:amd64 (255.4-1ubuntu8.10) ...
Setting up netplan.io (1.1.2-2~ubuntu24.04.2) ...
Setting up libpam-systemd:amd64 (255.4-1ubuntu8.10) ...
Setting up cloud-init (25.1.4-0ubuntu0~24.04.1) ...
Setting up apport-core-dump-handler (2.28.1-0ubuntu3.8) ...
Setting up apport (2.28.1-0ubuntu3.8) ...
apport-autoreport.service is a disabled or a static unit not running, not starting it.
Processing triggers for dbus (1.14.10-4ubuntu4.1) ...
Processing triggers for initramfs-tools (0.142ubuntu25.5) ...
update-initramfs: Generating /boot/initrd.img-6.11.0-1018-azure
Processing triggers for libc-bin (2.39-0ubuntu8.5) ...
Processing triggers for rsyslog (8.2312.0-3ubuntu9.1) ...
Processing triggers for ufw (0.36.2-6) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning candidates...
Scanning linux images...

Running kernel seems to be up-to-date.

Restarting services...
systemctl restart chrony.service cron.service multipathd.service packagekit.service
polkit.service udisks2.service walinuxagent.service

Service restarts being deferred:
systemctl restart ModemManager.service
/etc/needrestart/restart.d/dbus.service
systemctl restart getty@tty1.service
systemctl restart networkd-dispatcher.service
systemctl restart serial-getty@ttyS0.service
systemctl restart systemd-logind.service
systemctl restart unattended-upgrades.service

No containers need to be restarted.

User sessions running outdated binaries:
 felipeuser @ session #1: apt[2107], sshd[1588,1717]
 felipeuser @ user manager service: systemd[1593]

```

Ilustración 89 Actualización del sistema y reinicio de servicios. Autoría propia

Paso 31: Crear una carpeta dentro de la máquina virtual

Desde la terminal, puedes usar comandos como mkdir para crear carpetas.

En este caso, se creó una carpeta llamada micarpetacomputo dentro del directorio /home.

Esto demuestra que ya puedes comenzar a organizar tus archivos y proyectos directamente desde la máquina virtual.

```

No containers need to be restarted.

User sessions running outdated binaries:
 felipeuser @ session #1: apt[2107], sshd[1588,1717]
 felipeuser @ user manager service: systemd[1593]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
felipeuser@virtua-Machine:~$ sudo mkdir /home/micarpetacomputo
felipeuser@virtua-Machine:~$ ls
felipeuser@virtua-Machine:~$ ls -l /home
total 8
drwxr-x--- 4 felipeuser felipeuser 4096 Jul 30 20:41 felipeuser
drwxr-xr-x  2 root      root      4096 Jul 30 20:44 micarpetacomputo
felipeuser@virtua-Machine:~$ █

```

Ilustración 90 Creación de una carpeta desde la terminal. Autoría propia

Paso 32: Crear un nuevo usuario y su carpeta persona

Desde la terminal puedes crear nuevos usuarios con el comando sudo useradd.

En este caso se creó el usuario roman y se le asignó la carpeta /home/roman como directorio personal.

Se usó el comando chown para otorgarle permisos al nuevo usuario sobre esa carpeta.

Esto permite gestionar múltiples usuarios dentro de la misma máquina virtual.

```
felipeuser@virtua-Machine:~$ sudo useradd roman
felipeuser@virtua-Machine:~$ id roman
uid=1001(roman) gid=1001(roman) groups=1001(roman)
felipeuser@virtua-Machine:~$ sudo mkdir /home/david
felipeuser@virtua-Machine:~$ sudo chown roman:roman /home/roman
chown: cannot access '/home/roman': No such file or directory
felipeuser@virtua-Machine:~$ sudo mkdir /home/roman
sudo chown roman:roman /home/roman
felipeuser@virtua-Machine:~$ sudo chown roman:roman /home/roman
felipeuser@virtua-Machine:~$ id roman
uid=1001(roman) gid=1001(roman) groups=1001(roman)
felipeuser@virtua-Machine:~$ █
```

Ilustración 91 Alta de usuario y asignación de directorio propio. Autoría propia

Paso 33: Crear un grupo nuevo en la máquina virtual

Puedes crear grupos personalizados para organizar usuarios dentro del sistema. En este ejemplo se creó el grupo **grupo_901** con el comando sudo groupadd grupo_901. Luego, con getent group grupo_901, se verificó que el grupo fue creado correctamente.

```
felipeuser@virtua-Machine:~$ sudo groupadd grupo_901
felipeuser@virtua-Machine:~$ getent group grupo_901
grupo_901:x:1002:
felipeuser@virtua-Machine:~$ █
```

Ilustración 92 Alta de grupo en el sistema. Autoría propia

ALMACENAMIENTO

Paso 1: Buscar el servicio de cuentas de almacenamiento

Desde el portal de Azure, en la barra superior de búsqueda, escribe la palabra “cuentas”.

En los resultados aparecerá el servicio llamado Cuentas de almacenamiento dentro de la sección Servicios.

Haz clic sobre esa opción para comenzar a crear una nueva cuenta de almacenamiento.

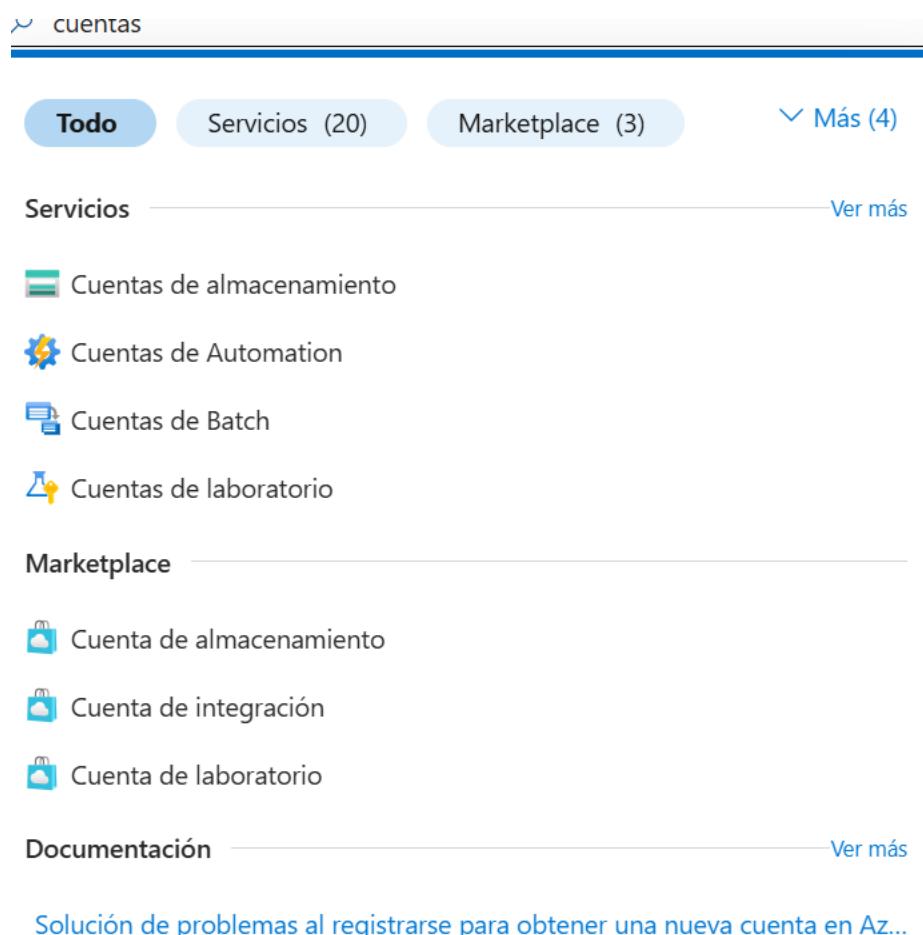


Ilustración 93 Ubicación del servicio “Cuentas de almacenamiento” en Azure. Autoría propia

Paso 2: Iniciar la creación de una cuenta de almacenamiento

Al ingresar al servicio, se muestra una lista de las cuentas de almacenamiento disponibles.

En este caso, no hay ninguna creada todavía, por lo que aparece el mensaje “No hay cuentas de almacenamiento para mostrar”.

Haz clic en el botón + Crear para comenzar el proceso de creación de una nueva cuenta.

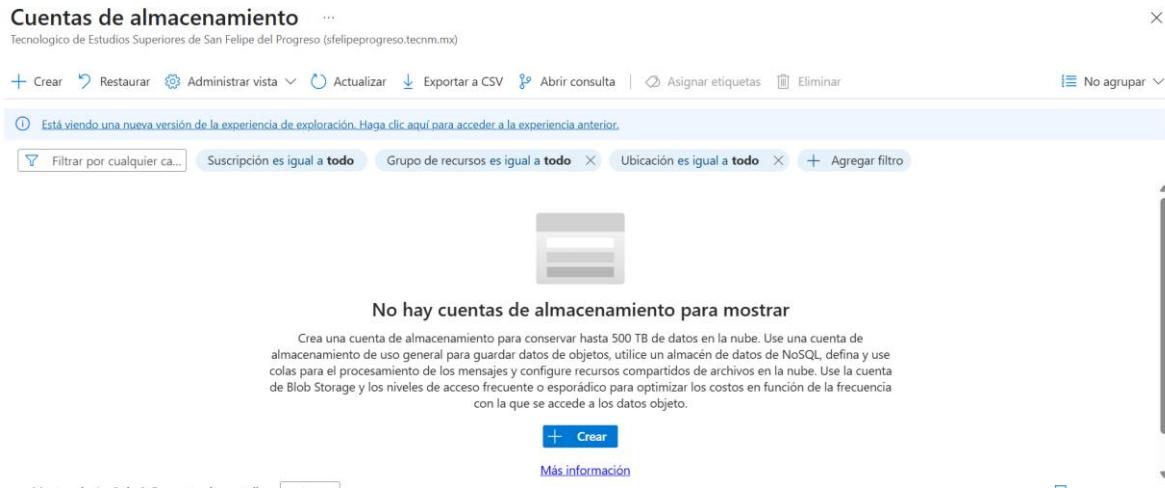


Ilustración 94 Pantalla inicial del servicio “Cuentas de almacenamiento”. Autoría propia

Paso 3: Crear una nueva cuenta de almacenamiento

En esta pantalla se configuran los detalles del proyecto y de la instancia para la cuenta de almacenamiento.

Crear una cuenta de almacenamiento

Detalles del proyecto

Seleccione la suscripción en la que se creará la nueva cuenta de almacenamiento. Elija un grupo de recursos nuevo o uno ya existente para organizar y administrar la cuenta de almacenamiento junto con otros recursos.

Suscripción *

Grupo de recursos *

 [Crear nuevo](#)

Detalles de la instancia

Nombre de la cuenta de almacenamiento *

(1)

Región * (1)

[Implementación en una zona extendida de Azure](#)

Servicio principal (1)

Rendimiento * (1)

Estándar: Opción recomendada para la mayoría de los escenarios (cuenta de uso general v2)

[Anterior](#)

[Siguiente](#)

Revisar y crear

Ilustración 95 Formulario de configuración de cuenta de almacenamiento en Azure. Autoría propia

Paso 4: Configuración avanzada de la cuenta de almacenamiento

En esta sección se ajustan detalles clave del almacenamiento:

- **Nombre de la cuenta de almacenamiento:** almacenare
- **Región:** (Mexico) Mexico Central
- **Rendimiento:** Estándar (adecuado para la mayoría de los casos generales)
- **Redundancia:** Almacenamiento con redundancia local (LRS), lo que significa que los datos se replican en un solo datacenter dentro de la región seleccionada.

Detalles de la instancia

Nombre de la cuenta de almacenamiento * ⓘ	almacenare
Región * ⓘ	(Mexico) Mexico Central <small>Implementación en una zona extendida de Azure</small>
Servicio principal ⓘ	Seleccionar un servicio principal
Rendimiento * ⓘ	<input checked="" type="radio"/> Estándar: Opción recomendada para la mayoría de los escenarios (cuenta de uso general v2) <input type="radio"/> Prémium: Se recomienda para escenarios que requieren una latencia baja.
Redundancia * ⓘ	Almacenamiento con redundancia local (LRS)

[Anterior](#) [Siguiente](#) [Revisar y crear](#)

Ilustración 96 Selección de región, rendimiento y redundancia. Autoría propia

Paso 5: Revisión final de la cuenta de almacenamiento

En esta pantalla se muestra un resumen completo de la configuración realizada antes de crear la cuenta de almacenamiento.

Datos básicos Avanzado Redes Protección de datos Cifrado Etiquetas **Revisar y crear**

[Ver plantilla de automatización](#)

Datos básicos

Suscripción	Azure for Students
Grupo de recursos	AzureDFR
Ubicación	West US 3
Nombre de la cuenta de almacenamiento	almacenare
Servicio principal	
Rendimiento	Estándar
Replicación	Almacenamiento con redundancia local (LRS)

Avanzado

Habilitar el espacio de nombres jerárquico	Deshabilitado
Habilitar SFTP	Deshabilitado

[Anterior](#)

[Siguiente](#)

Crear

Ilustración 97 Resumen de configuración antes de crear la cuenta de almacenamiento. Autoría propia

Paso 6: Implementación en curso

Una vez revisada y aprobada la configuración, el sistema de Azure comienza la implementación automática de la cuenta de almacenamiento en la nube. En este paso, se muestra una notificación emergente indicando que se está inicializando la implementación dentro del grupo de recursos "AzureDFR". Esta fase puede tomar algunos segundos mientras se aprovisiona la infraestructura solicitada.

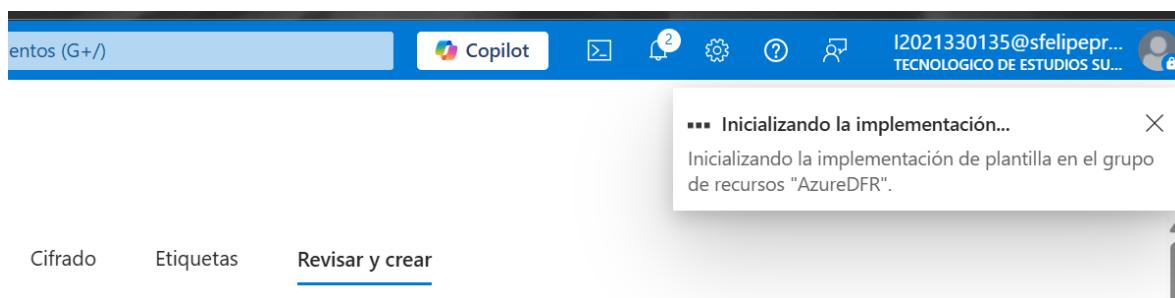


Ilustración 98 Inicializando la implementación de la cuenta de almacenamiento. Autoría propia

Paso 7: Supervisión de la implementación de la cuenta de almacenamiento

Tras confirmar la creación del recurso, Azure inicia la implementación de la cuenta de almacenamiento en el grupo de recursos asignado. En esta etapa, se muestra el estado del proceso en tiempo real, junto con la hora de inicio, el ID de correlación y el grupo de recursos.

The screenshot shows the Azure portal interface for monitoring a storage account implementation. At the top, there are buttons for 'Eliminar' (Delete), 'Cancelar' (Cancel), 'Volver a implementar' (Reimplement), 'Descargar' (Download), and 'Actualizar' (Update). Below this, a section titled 'La implementación está en curso' (The implementation is in progress) displays the following details:

- Nombre de implementación: almacenare_175...
- Suscripción: Azure for Students
- Grupo de recursos: AzureDFR
- Hora de inicio: 30/7/2025, 2:59:19 p.m.
- Id. de correlación: cd34cd3e-3f33-4791-a2f3-25233b95

Below this, a section titled 'Detalles de implementación' (Implementation details) shows a table with columns 'Recurso', 'Tipo', 'Estado', and 'Detalles de la opera...'. A note states 'No hay ningún resultado.' (No results found).

At the bottom, there are links for 'Enviar comentarios' (Send comments) and 'Cuéntenos su experiencia con la implementación' (Tell us about your implementation experience).

Ilustración 99 Implementación en curso en Azure. Autoría propia

Paso 8: Confirmación de implementación exitosa

Una vez finalizada la implementación de la cuenta de almacenamiento, Azure muestra una notificación de confirmación indicando que el proceso se completó exitosamente. En esta vista, se puede acceder directamente al recurso creado mediante el botón "Ir al recurso"

The screenshot shows the Azure portal interface for confirming a storage account implementation. At the top, there are buttons for 'Buscar recursos, servicios y documentos (G/)' (Search resources, services and documents), 'Copilot', and user information 'I2021330135@stelipepr... TECNOLOGICO DE ESTUDIOS SU...'.

The main content area shows the implementation status for 'almacenare_1753909136890' with a green checkmark icon and the message 'Se completó la implementación' (The implementation was completed). It also lists the implementation details:

- Nombre de implementación: almacenare_175...
- Suscripción: Azure for Students
- Grupo de recursos: AzureDFR
- Hora de inicio: 30/7/2025, 2:59:19 p.m.
- Id. de correlación: cd34cd3e-3f33-4791-a2f3-25233b95

Below this, there are sections for 'Detalles de implementación' (Implementation details) and 'Pasos siguientes' (Next steps). A 'Ir al recurso' (Go to resource) button is present.

A modal window on the right provides additional information and links:

- Implementación correcta**: La implementación "almacenare_1753909136890" se realizó correctamente en el grupo de recursos "AzureDFR".
- Ir al recurso** (button)
- Anclar al panel** (button)
- Cost Management**: Obtenga una notificación para permanecer dentro del presupuesto y evitar cargos inesperados en su factura. [Configurar alertas de costo >](#)
- Seguridad** (icon)

Ilustración 100 Implementación completada correctamente. Autoría propia

Paso 9: Verificación de la cuenta de almacenamiento

Esta pantalla muestra los detalles esenciales de la cuenta de almacenamiento recién creada.

Cargar		Abrir en el Service Fabric Explorer	Eliminar	Mover	Actualizar	Abrir en dispositivos móviles	CLI/PS	...
Vista JSON								
^	Essentials							
Grupo de recursos	(mover)	Rendimiento						
AzureDFR		Estándar						
Ubicación		Replicación						
westus3		Almacenamiento con redundancia local (LRS)						
Suscripción	(mover)	Tipo de cuenta						
Azure for Students		StorageV2 (uso general v2)						
Id. de suscripción		Estado de aprovisionamiento						
ddfd0fef-065d-4cde-aabf-6c11c91814a2		Correcto						
Estado del disco		Creada						
Disponible		30/7/2025, 2:59:22 p.m.						
Etiquetas	(editar)							
	Agregar etiquetas							

Ilustración 101 Vista general del recurso creado. Autoría propia

Paso 10: Subida de archivos al almacenamiento (Blob Storage)

Esta interfaz permite cargar archivos a la cuenta de almacenamiento tipo **Blob** en Azure.

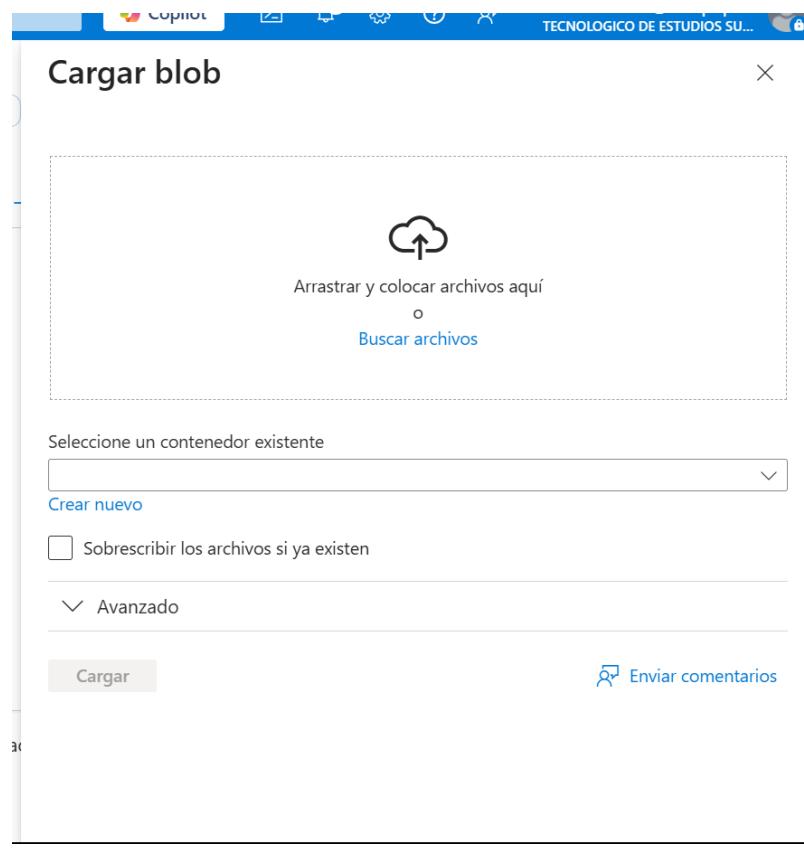


Ilustración 102 Cargar archivos al blob. Autoría propia

Paso 11: Creación de contenedor en Blob Storage

En este paso se está creando un nuevo contenedor llamado documentos dentro del servicio de almacenamiento Blob en Azure.

Se especifica el nivel de acceso anónimo como Privado, lo que significa que solo usuarios autenticados podrán acceder a los archivos del contenedor.

Nuevo contenedor

Nombre

documentos

Nivel de acceso anónimo (i)

Privado (sin acceso anónimo)

(i) El acceso anónimo a este contenedor se está bloqueando porque el acceso anónimo está deshabilitado en esta cuenta de almacenamiento.

Aceptar

Cancelar

Ilustración 103 Crear nuevo contenedor. Autoría propia

Paso 12: Confirmación de creación del contenedor "documentos"

Después de definir el nombre y nivel de acceso, se muestra un mensaje de confirmación:

"El contenedor de almacenamiento 'documentos' se creó correctamente."



Ilustración 104 Confirmación de contenedor creado. Autoría propia

Paso 13: Seleccionar archivo y preparar carga

Una vez creado el contenedor llamado "documentos", se selecciona el archivo que se desea cargar, en este caso:

"Practicas 1.0 David Felipe Roman.pdf".



Seleccione un contenedor existente

documentos

[Crear nuevo](#)

Sobreescritura de archivos si ya existen

▼ Avanzado

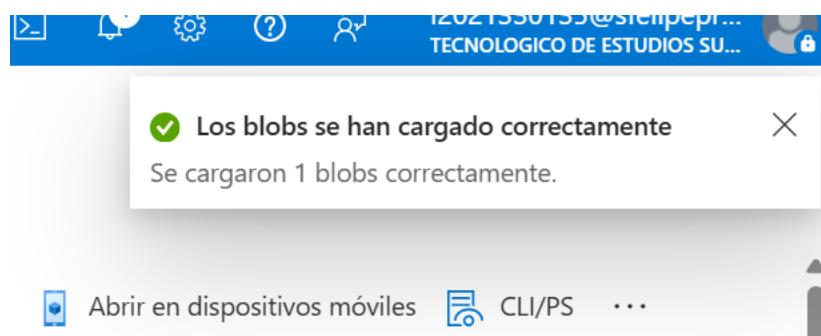
[Cargar](#)

Enviar comentarios

Ilustración 105 Archivo seleccionado para carga. Autoría propia

Paso 14: Confirmación de que el archivo se cargó correctamente

Después de hacer clic en el botón “Cargar”, el sistema muestra un mensaje de confirmación



Abrir en dispositivos móviles CLI/PS ...

Ilustración 106 Confirmación de carga exitosa. Autoría propia

Paso 15: Acceder a la sección de Contenedores en Azure

Desde el portal de Microsoft Azure, en el menú lateral izquierdo, selecciona la opción “Almacenamiento de datos” y haz clic en “Contenedores”.

Esta sección te permite administrar los blobs (archivos) dentro de una cuenta de almacenamiento previamente creada.

The screenshot shows the Azure Storage Explorer interface. On the left, there's a sidebar with icons for Events, Storage Explorer, Storage Mover, Partner Solutions, Resource Visualizer, and Data Storage. Under Data Storage, 'Almacenamiento de datos' is expanded, showing 'Contenedores' (Containers) and 'Recursos compartidos' (Shared Resources). The 'Contenedores' tab is selected, showing a list with one item: 'documentos'. The main pane on the right displays general storage properties: 'Id. de suscripción' (Subscription ID) is 'ddfd0fef-065d-4cde-aabf-6c11c918'; 'Estado del disco' (Disk status) is 'Disponible' (Available); and there are tabs for 'Etiquetas (editar)' (Labels edit) and 'Agregar etiquetas' (Add labels). Below these are 'Propiedades' (Properties) and 'Supervisión' (Monitoring) tabs, with 'Propiedades' currently selected. At the bottom, a URL bar shows the full URL of the page.

Ilustración 107 Navegación a Contenedores. Autoría propia

Paso 16: Verificar la carga del archivo en el contenedor

Una vez finalizada la carga, accede al contenedor llamado "**documentos**" y verifica que el archivo haya sido cargado correctamente. En este caso, el archivo Practicas 1.0 David Felipe Roman.pdf aparece con su fecha de modificación, tamaño, tipo de blob y estado de disponibilidad.

The screenshot shows the contents of the 'documentos' container. At the top, there's a toolbar with buttons for 'Agregar directorio' (Create directory), 'Cargar' (Upload), 'Cambiar nivel de acceso' (Change access level), 'Actualizar' (Update), 'Eliminar' (Delete), 'Copiar' (Copy), 'Pegar' (Paste), 'Cambiar nombre' (Change name), and more. Below the toolbar, the container name 'documentos' is listed. A message indicates the authentication method: 'Método de autenticación: Clave de acceso (Cambiar a la cuenta de usuario de Microsoft Entra)'. There are filters for 'Agregar filtro' (Add filter) and 'Buscar blobs por prefijo (distingue mayúsculas de minúsculas)' (Search blobs by prefix (distinguishes uppercase from lowercase)). A dropdown menu 'Solo mostrar blobs activos' (Show only active blobs) is open. The main area shows a table with the following data:

Nombre	Última modificación	Nivel de acceso	Tipo de blob	Tamaño	Estado de con...
Practicas 1.0 Davi...	30/7/2025, 3:02:55 p.m.	Frecuente (inferido)	Blob en bloques	2.4 MiB	Disponible

Ilustración 108 Visualización del archivo cargado en el contenedor. Autoría propia

Paso 17: Ver información detallada del archivo

Haz clic sobre el archivo dentro del contenedor para consultar sus propiedades. Aquí puedes ver:

- **URL del archivo** para acceso (si es público o compartido mediante SAS).
- **Tamaño y tipo** (application/pdf).
- **Fecha de modificación y creación.**
- **Nivel de acceso, estado, y cifrado.**

The screenshot shows the 'Información general' (General) tab of a blob's properties in Azure Storage. The URL is https://almacenare.blob... The file was last modified on 30/7/2025, 3:02:55 p.m. and created on the same date. It is a 2.4 MiB block blob with frequent access inferred. The access level is N/D. The file is not sealed or encrypted. It has a direct write rule and is not cached. The content type is application/pdf.

Propiedad	Valor
URL	https://almacenare.blob...
ÚLTIMA MODIFICACIÓN	30/7/2025, 3:02:55 p.m.
HORA DE CREACIÓN	30/7/2025, 3:02:55 p.m.
ID. DE VERSIÓN	-
TIPO	Blob en bloques
TAMAÑO	2.4 MiB
NIVEL DE ACCESO	Frecuente (inferido)
ÚLTIMA MODIFICACIÓN DEL NIVEL DE ACCESO	N/D
ESTADO DEL ARCHIVO	-
PRIORIDAD DE REHIDRATACIÓN	-
SERVIDOR CIFRADO	true
ETAG	0x8DDCFAC74C7730E
DIRECTIVA DE INMUTABILIDAD DE NIVEL DE VERSIÓN	Deshabilitado
CONTROL DE CACHÉ	[empty]
TIPO DE CONTENIDO	application/pdf

Ilustración 109 Propiedades del archivo cargado. Autoría propia

ANÁLISIS

La práctica de máquina virtual permitió simular un equipo físico desde la nube, configurado y operado remotamente. Este entorno virtualizado se puede utilizar para ejecutar software, realizar pruebas o mantener sistemas activos sin necesidad de hardware físico local. La facilidad para crear, iniciar, detener y eliminar la VM ofrece una experiencia real de administración de servidores en la nube.

El almacenamiento en blobs es ideal para guardar archivos de forma organizada y segura. Se creó un contenedor donde se cargó un archivo PDF, observando cómo Azure gestiona propiedades como tipo de contenido, tamaño, cifrado y acceso. Este tipo de almacenamiento es muy utilizado en aplicaciones web, respaldos y gestión documental en línea.

PRACTICA 3

INTRODUCCIÓN:

La transferencia de archivos entre una máquina física (local) y una máquina virtual (VM) en la nube, como las proporcionadas por Microsoft Azure, es una tarea fundamental para la gestión de datos y la implementación de aplicaciones. Para facilitar esta operación, se puede utilizar WinSCP, un cliente SFTP, SCP y FTP que permite una conexión segura y eficiente con sistemas remotos.

Este manual describe el proceso completo para conectar una máquina local con una máquina virtual Ubuntu en Azure, configurar WinSCP para la transferencia de archivos y verificar que los archivos se hayan transferido correctamente. A través de esta guía, los usuarios aprenderán a gestionar archivos de forma remota, con la ventaja de contar con una interfaz gráfica fácil de usar y un protocolo seguro de transferencia.

Requisitos:

Para realizar correctamente la transferencia de archivos entre la máquina física y la máquina virtual, se requieren los siguientes elementos y configuraciones previas:

Acceso a la Máquina Virtual en Azure:

- Una máquina virtual con **Ubuntu** instalada en Azure.
- La **IP pública** de la máquina virtual.
- El **usuario** y **contraseña** o la **clave SSH** configurada para la autenticación.

WinSCP Instalado:

- Descargar e instalar **WinSCP** desde su página oficial.
- Tener la clave privada **.ppk** o las credenciales de usuario listas para configurar la conexión con la máquina virtual.

Conexión SSH (SFTP):

- Asegurarse de que la máquina virtual tiene el **puerto 22 abierto** para conexiones SSH.
- Configurar la conexión en WinSCP utilizando el protocolo **SFTP** o **SCP**, especificando la IP, usuario y clave pública/contraseña.

CONEXIÓN REMOTA Y TRANSFERENCIA DE ACHIVOS A MV DE AZURE

Paso 1: Accedemos a la página oficial de WinSCP

<https://winscp.net>

Se muestra la página oficial de WinSCP en la sección de descargas, donde es posible obtener la versión más reciente del software (en este caso, la 6.5). También se observa un anuncio publicitario.

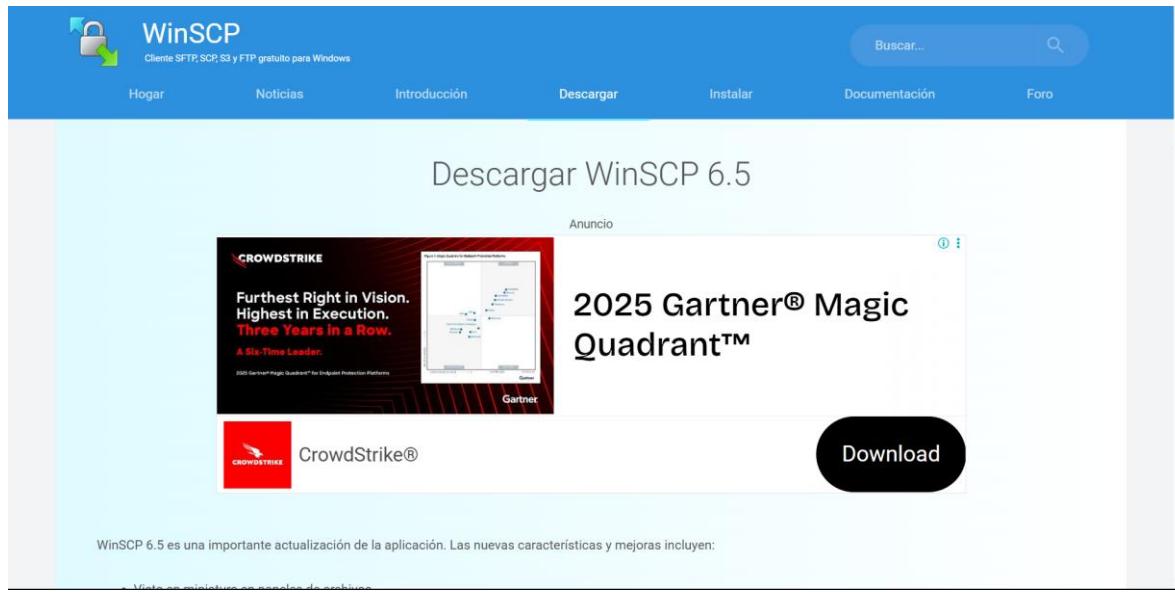


Ilustración 110 Página oficial de descarga de WinSCP. Autoría propia

Paso 2: Espera a que inicie la descarga

La imagen muestra la pantalla que aparece al hacer clic en el botón de descarga del instalador de WinSCP. Se indica que el archivo WinSCP-6.5.3-Setup.exe comenzará a descargarse automáticamente.

La descarga de WinSCP-6.5.3-Setup.exe comenzará en breve...

Sumas de comprobación

Anuncio

Anuncio

Ilustración 111 Inicio automático de la descarga de WinSCP. Autoría propia

Paso 3: Aceptar el acuerdo de licencia

Leemos los términos del acuerdo de licencia y haz clic en el botón “Aceptar” para continuar con la instalación del programa.

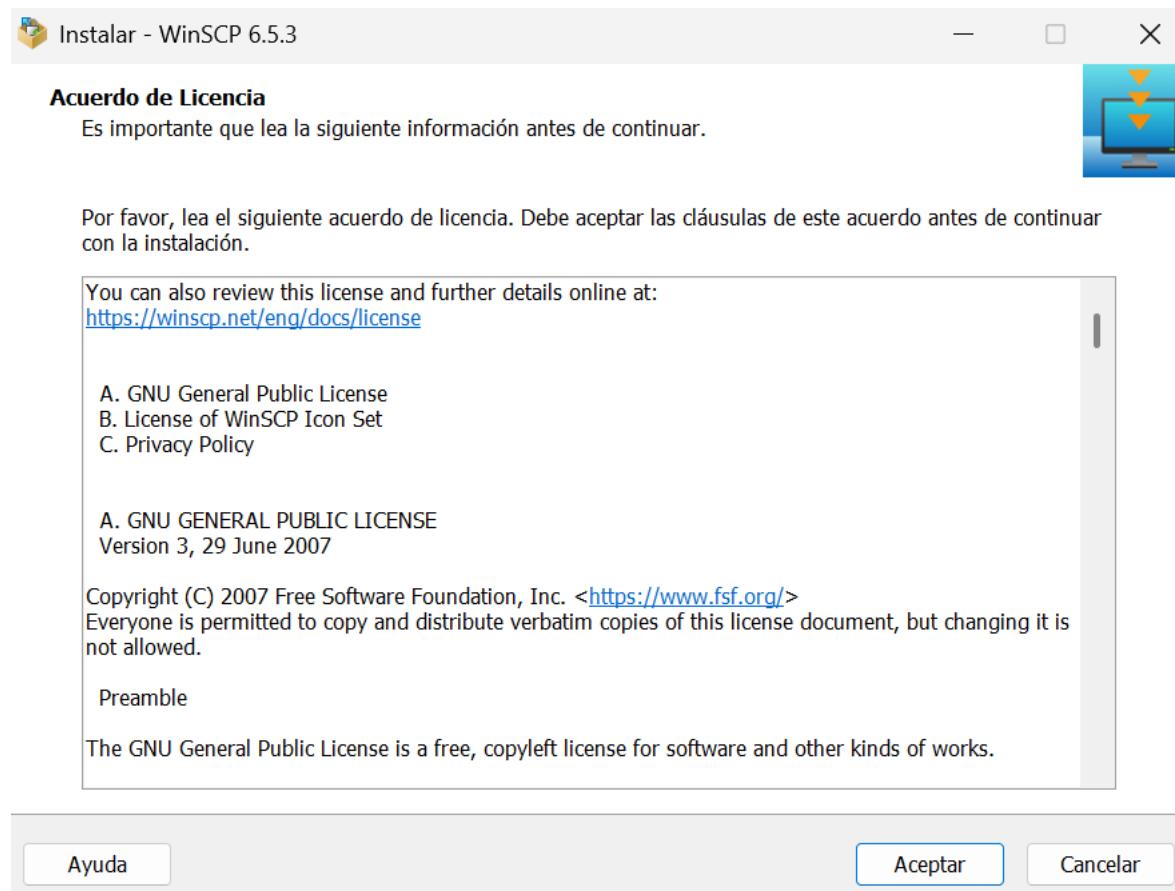


Ilustración 112 Acuerdo de Licencia de WinSCP. Autoría propia

Paso4: Seleccionar el tipo de instalación
Selecciona la opción "**Instalación típica (recomendada)**" para proceder con los ajustes predeterminados. Luego, haz clic en el botón "**Siguiente**".

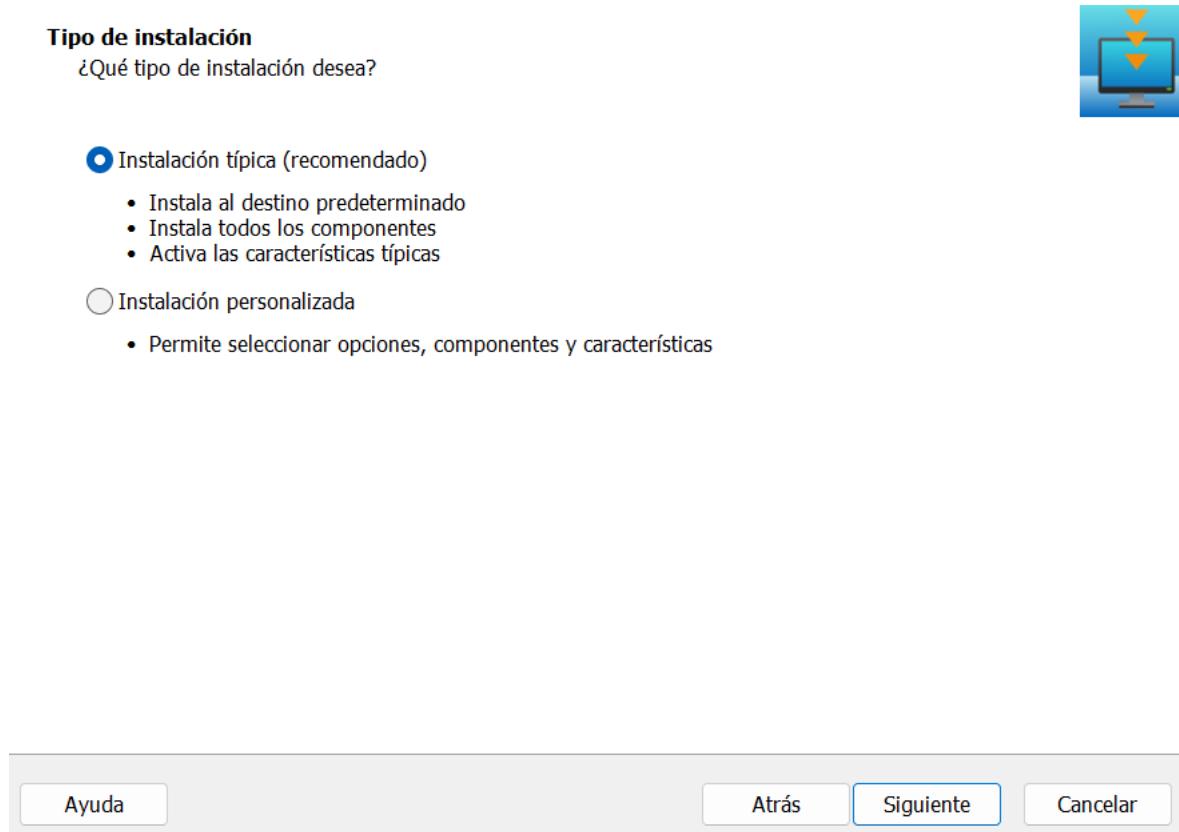


Ilustración 113 Selección del tipo de instalación en WinSCP. Autoría propia

Paso 5: Elegir la apariencia de la interfaz

La imagen muestra la ventana de configuración inicial de WinSCP, donde se debe elegir el estilo de interfaz. Se ofrecen dos opciones:

NC: con dos paneles (local y remoto) y atajos tipo Norton Commander.

Explorador: más similar al Explorador de Windows, mostrando solo el directorio remoto.

Configuración inicial

¿Qué apariencia prefiere?



Escoja el estilo de interfaz deseado.

NC



- dos paneles (local a la izquierda y remoto a la derecha)
- atajos de teclado como NC y otros programas similares (Total Commander, Midnight Commander,...)
- «arrastrar y soltar» entre los dos paneles

Explorador



- solo el directorio remoto
- atajos de teclado como en el Explorador de Windows
- «arrastrar y soltar»

Ayuda

Atrás

Siguiente

Cancelar

Ilustración 114 Selección de apariencia inicial en WinSCP. Autoría propia

Paso 6: Confirmar y comenzar la instalación

Revisa los componentes y configuraciones seleccionadas. Si estás de acuerdo, haz clic en el botón “Instalar” para iniciar la instalación de WinSCP en tu equipo.

Listo para Instalar

Ahora el programa está listo para iniciar la instalación de WinSCP en su sistema.



Haga clic en Instalar para continuar con el proceso o haga clic en Atrás si desea revisar o cambiar alguna configuración.

Componentes Seleccionados:

- Ejecutable WinSCP
- Extensión para «arrastrar y soltar» (requiere reiniciar)
- Pageant (agente de autenticación SSH)
- PuTTYgen (generador de claves)
- Traducciones

Tareas Adicionales:

- Buscar automáticamente actualizaciones (recomendado)
- Recolectar estadísticas de uso anónimas
- Crear ícono en el escritorio
- Añadir enlace al menú contextual «Enviar a» del Explorador
- Registrar para manejar direcciones URL

Configuración inicial (se puede cambiar después)

NC

Ayuda

Atrás

Instalar

Cancelar

Ilustración 115 Resumen final antes de instalar WinSCP. Autoría propia

Paso 7: Esperar mientras se instala el programa

Espera unos momentos mientras el sistema instala WinSCP en tu equipo. No cierres esta ventana hasta que el proceso finalice y se confirme la instalación exitosa.

Instalando

Por favor, espere mientras se instala WinSCP en su sistema.



Buscando sitios para importar...

Ilustración 116 Instalación en progreso de WinSCP. Autoría propia

Paso 8: Importar sesiones desde PuTTY (opcional)

Si tienes sesiones previamente guardadas en PuTTY, haz clic en “Sí” para importarlas a WinSCP. Si prefieres hacerlo después, selecciona “No”. Puedes acceder a esta opción más tarde desde el menú de conexión.

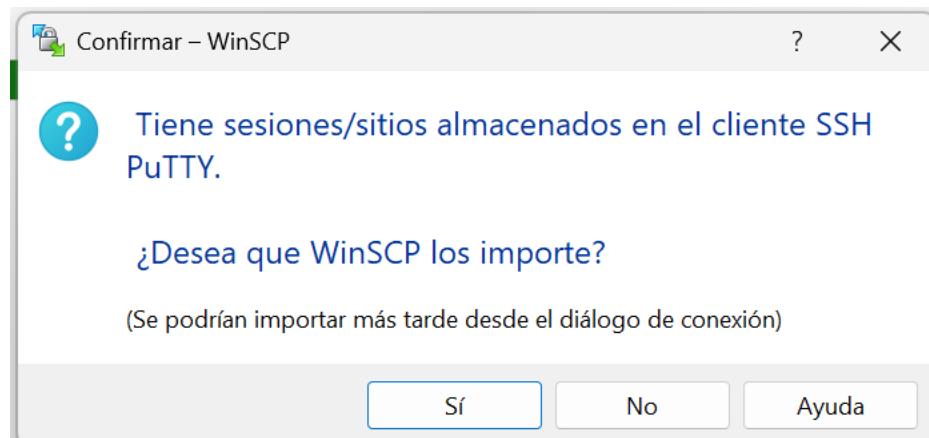


Ilustración 117 Importar sesiones almacenadas de PuTTY a WinSCP. Autoría propia

Paso 9: Seleccionar sesiones para importar

Marca las sesiones que deseas importar desde PuTTY (por defecto están seleccionadas). Luego, haz clic en “Aceptar” para agregarlas a tu lista de conexiones disponibles en WinSCP.

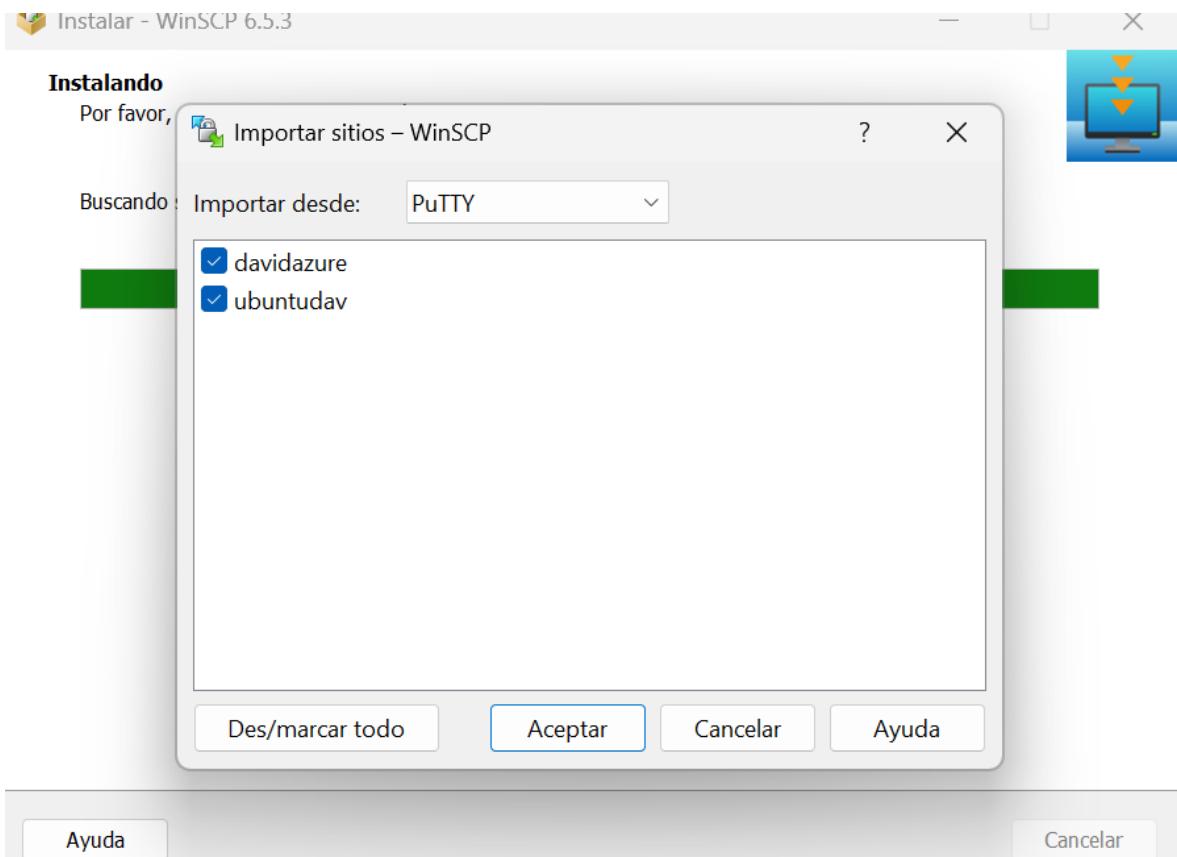


Ilustración 118 Selección de sitios para importar desde PuTTY. Autoría propia

Paso 10: Finalizar la instalación e iniciar WinSCP

Deja marcadas las opciones “Iniciar WinSCP” y, si lo deseas, “Abrir sitio web de ayuda”. Luego haz clic en “Finalizar” para completar la instalación y lanzar la aplicación.



Ilustración 119 Finalización de la instalación de WinSCP. Autoría propia

Paso 11: Configurar la conexión a la máquina virtual

En la sección de "Nombre o IP del servidor", introduce la IP pública de tu máquina virtual en Azure (por ejemplo, 20.151.57.144).

En "Usuario", ingresa el nombre de usuario de la VM (como felipeuser).

En "Contraseña", pon la contraseña correspondiente (si usas clave SSH, no es necesario poner contraseña aquí).

Luego, haz clic en "Conectar".

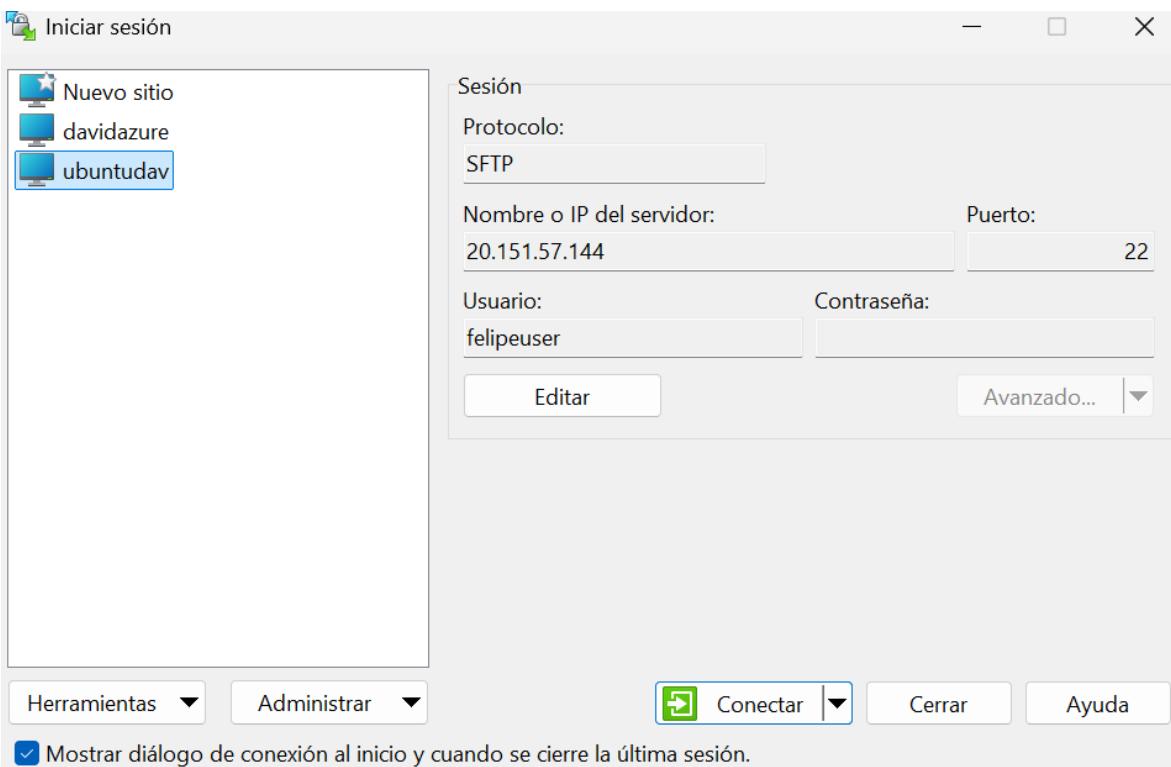


Ilustración 120 Configuración de conexión en WinSCP a la máquina virtual de Azure. Autoría propia

Paso 12: Ingresar la contraseña de la clave SSH

Si la conexión utiliza una clave privada para autenticarte, ingresa la contraseña de la clave en el campo correspondiente. Luego haz clic en “Aceptar” para completar la autenticación y establecer la conexión.

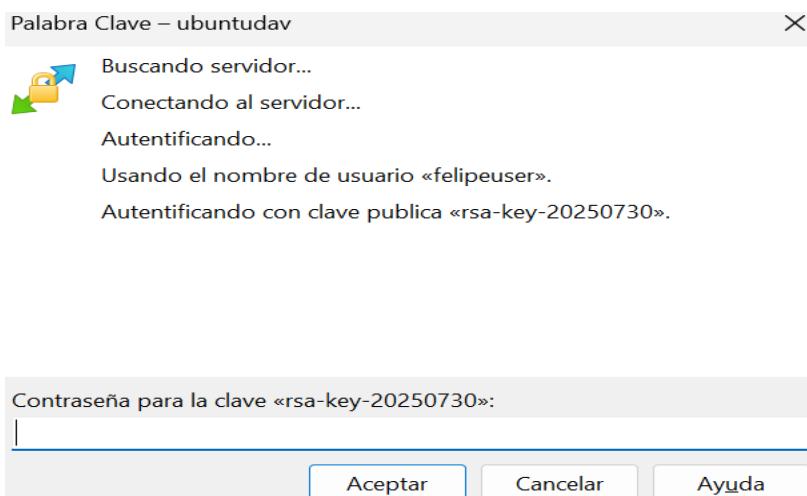


Ilustración 121 Autenticación con clave pública en WinSCP. Autoría propia

Paso 13: Transferir archivos entre la máquina local y la VM

Puedes arrastrar y soltar archivos entre ambos paneles. El panel izquierdo muestra los archivos locales de tu computadora, y el panel derecho muestra los archivos de la máquina virtual en Azure. Copia los archivos que necesites entre ambos lugares.

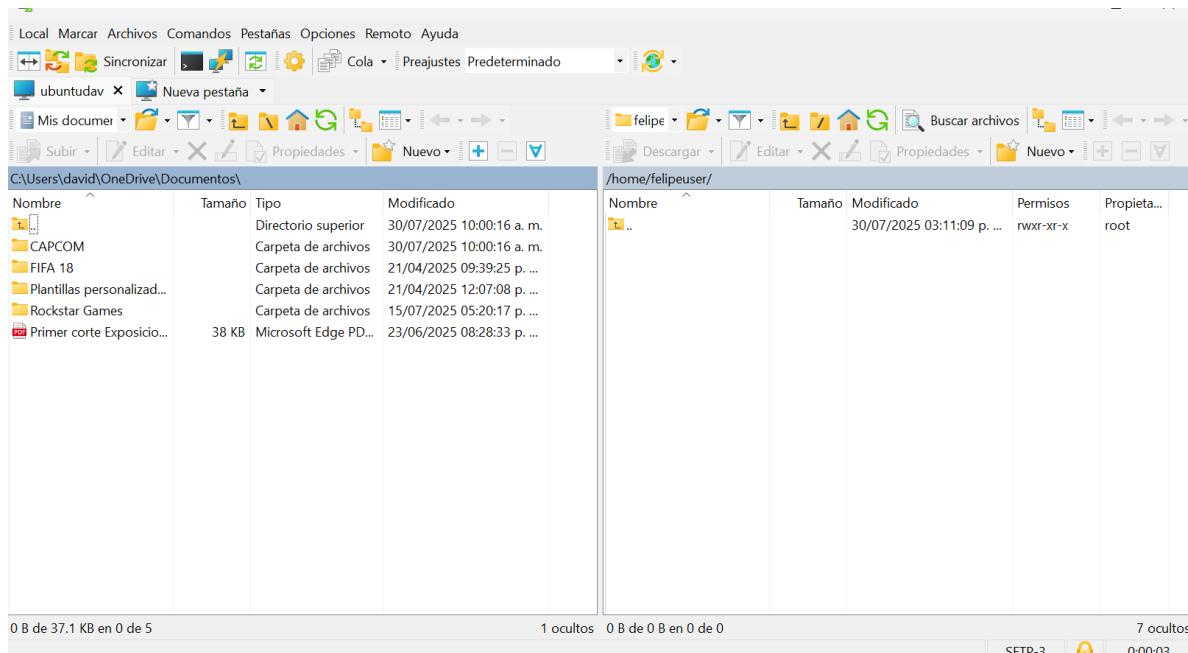


Ilustración 122 Interfaz de WinSCP: Conexión establecida con la máquina virtual. Autoría propia

Paso 14: Subir archivos a la máquina virtual

Para subir archivos a tu máquina virtual, selecciona el archivo desde el panel izquierdo (local) y arrástralo al directorio deseado en el panel derecho (remoto). También puedes hacer clic derecho sobre el archivo y seleccionar la opción "Subir".

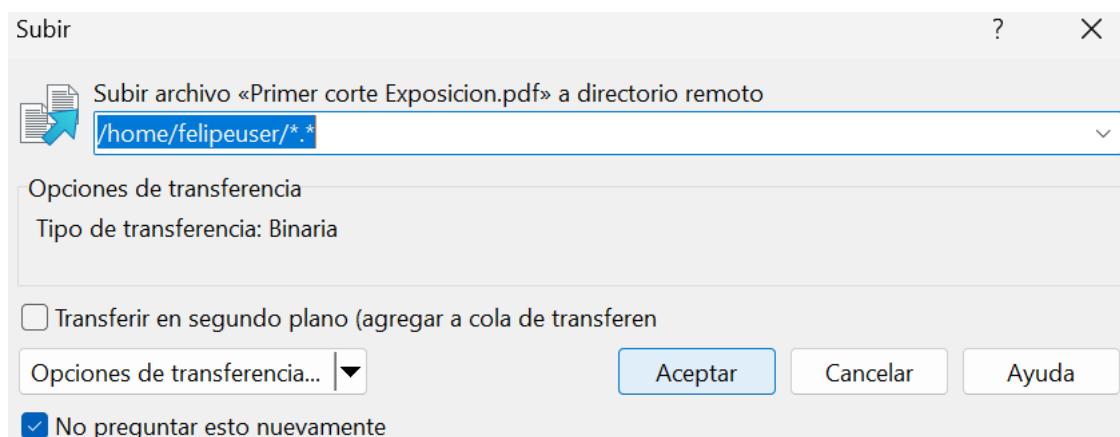


Ilustración 123 Subir archivo desde la máquina local a la máquina virtual. Autoría propia

Paso 15: Confirmación de transferencia completada

Una vez que la transferencia se haya completado, WinSCP notificará que el archivo ha sido subido con éxito. Puedes continuar con más transferencias o cerrar la sesión cuando termines.

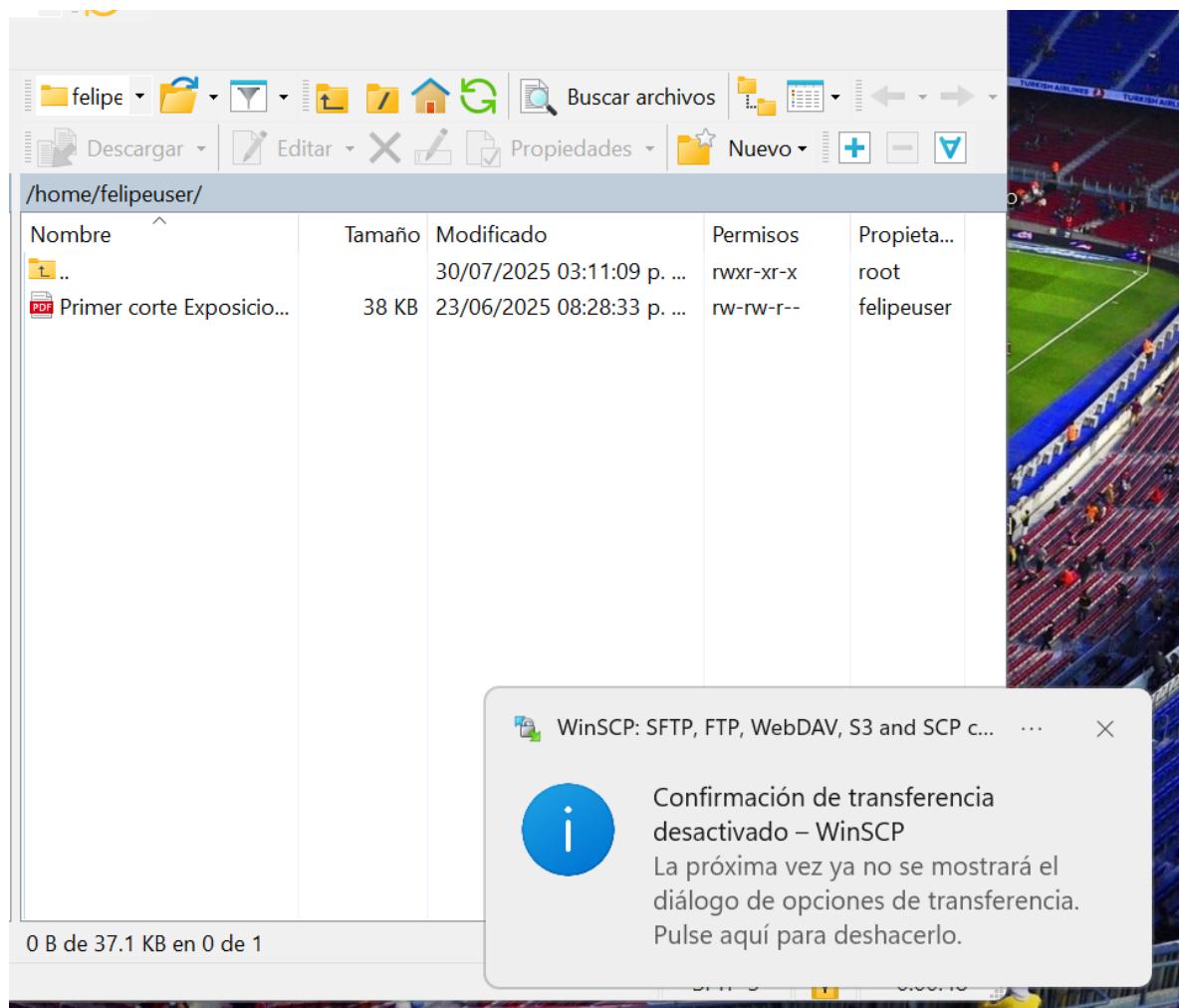
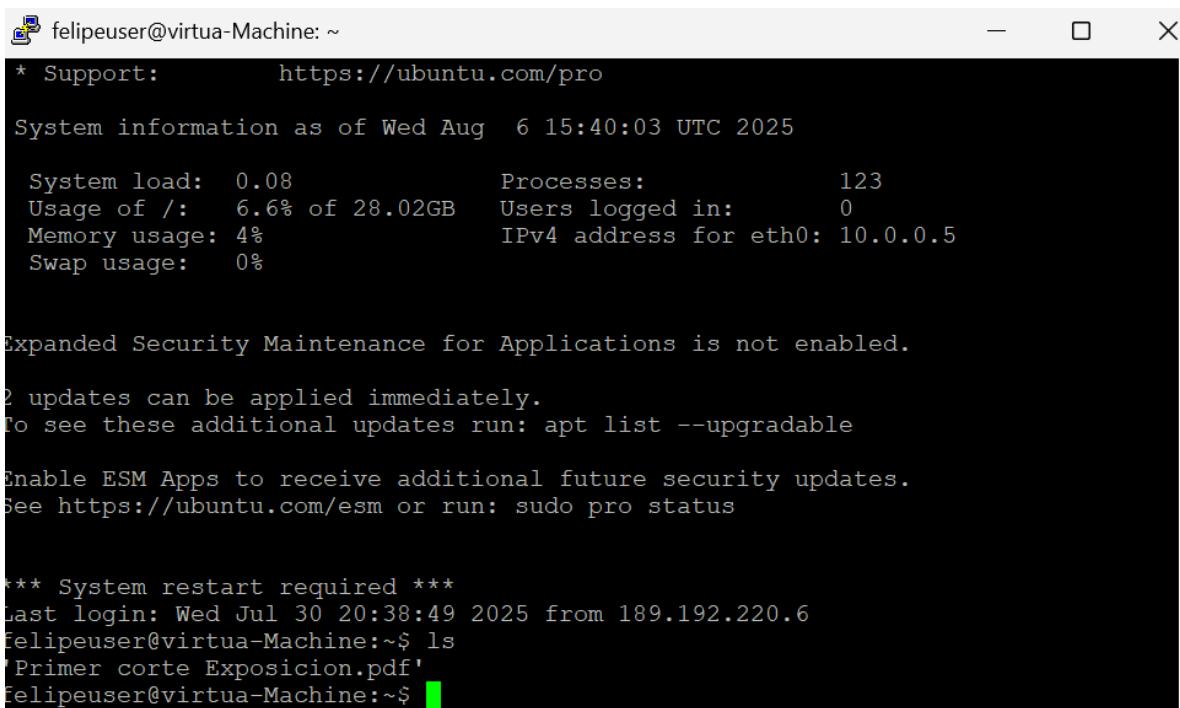


Ilustración 124 Confirmación de la transferencia exitosa de archivo en WinSCP. Autoría propia

Paso 16: Verificar archivo en la VM

Para verificar que el archivo se haya transferido correctamente, usa el comando ls en la terminal de Ubuntu. Si el archivo aparece en el listado, significa que la transferencia fue exitosa y el archivo está disponible en la VM.



The screenshot shows a terminal window titled "felipeuser@virtua-Machine: ~". It displays system information as of Wednesday, August 6, 2025, at 15:40:03 UTC. The system load is 0.08, processes are 123, and memory usage is 4%. It also shows swap usage at 0% and an IPv4 address for eth0 as 10.0.0.5. A message indicates that expanded security maintenance for applications is not enabled, and 2 updates can be applied immediately. It suggests enabling ESM Apps for future security updates. The terminal then shows a command-line session where the user runs "ls" and lists a file named "Primer corte Exposicion.pdf".

```
* Support: https://ubuntu.com/pro
System information as of Wed Aug  6 15:40:03 UTC 2025
System load: 0.08          Processes: 123
Usage of /: 6.6% of 28.02GB Users logged in: 0
Memory usage: 4%          IPv4 address for eth0: 10.0.0.5
Swap usage: 0%
Expanded Security Maintenance for Applications is not enabled.
2 updates can be applied immediately.
To see these additional updates run: apt list --upgradable
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

*** System restart required ***
Last login: Wed Jul 30 20:38:49 2025 from 189.192.220.6
felipeuser@virtua-Machine:~$ ls
'Primer corte Exposicion.pdf'
felipeuser@virtua-Machine:~$
```

Ilustración 125 Confirmación de archivo transferido en la terminal de Ubuntu. Autoría propia

ANALISIS

La transferencia de archivos entre sistemas locales y remotos es una actividad esencial en entornos de desarrollo y producción, especialmente cuando se trabajan con máquinas virtuales. El uso de protocolos como SFTP (Secure File Transfer Protocol) asegura que los archivos se transfieran de manera segura, cifrada y sin riesgos de intervención externa.

En este caso, WinSCP es la herramienta elegida por su sencillez, versatilidad y soporte para múltiples protocolos, permitiendo transferencias rápidas y seguras. Además, WinSCP ofrece una interfaz gráfica que facilita la interacción con la máquina remota a través de dos paneles (local y remoto), permitiendo arrastrar y soltar archivos con facilidad.

El análisis de este proceso resalta la importancia de seguir los pasos de autenticación correctamente (usando claves SSH o contraseñas) y asegurar que las configuraciones de conexión estén optimizadas para el acceso remoto sin fallos.



TECNOLÓGICO
NACIONAL DE MÉXICO

TESSFP
TECNOLÓGICO DE ESTUDIOS SUPERIORES
SAN FELIPE DEL PROGRESO

TECNOLÓGICO DE ESTUDIOS SUPERIORES DE SAN FELIPE DEL PROGRESO

CÓMPUTO EN LA NUBE

PRACTICAS 3.3

DOCENTE:

LIZETH JOSE ANGELES

ALUMNO:

DAVID FELIPE ROMAN

INGENIERÍA INFORMÁTICA

NOVENO SEMESTRE

INDICE

INTRODUCCIÓN.....	5
REQUERIMIENTOS	5
CREAR REPOSITORIO	7
INICIAR SESION VERCCEL.....	11
IMPORTAR PROYECTO DE GITHUB A VERCCEL	13
INICIAR SESION EN SUPABASE Y VINCULAR CON GITHUB.....	19
AUTENTICACION.....	22
BASES DE DATOS.....	29
SUBIR ARCHIVOS.....	36
ACTUALIZAR Y ELIMINAR REGISTROS.....	42
Análisis de la práctica	51

Ilustración 1 Creación de un nuevo repositorio en GitHub	7
Ilustración 2 Clonar un repositorio desde GitHub Desktop	8
Ilustración 3 Abrir el repositorio en Visual Studio Code	8
Ilustración 4 Visual Studio Code con el repositorio abierto.....	9
Ilustración 5 Creación de la estructura HTML del proyecto	10
Ilustración 6 Guardar cambios y realizar commit en GitHub Desktop.....	10
Ilustración 7 Iniciar sesión en Vercel.....	11
Ilustración 8 Autorizar permisos de GitHub en Verce	12
Ilustración 9 Panel principal de Vercel	12
Ilustración 10 Importar un proyecto en Vercel.....	13
Ilustración 11 Seleccionar proveedor para importar repositorio.....	13
Ilustración 12 Conceder permisos de acceso a repositorios en Vercel.....	14
Ilustración 13 Confirmar acceso en GitHub.....	14
Ilustración 14 Seleccionar repositorio para importar en Vercel.....	15
Ilustración 15 Configuración del nuevo proyecto en Vercel.....	16
Ilustración 16 Proyecto implementado con éxito en Vercel	17
Ilustración 17 Vista del despliegue de producción en Vercel.....	18
Ilustración 18 Proyecto funcionando en dominio de Vercel	18
Ilustración 19 Página principal de Supabase	19
Ilustración 20 Iniciar sesión en Supabase	20
Ilustración 21 Autorizar acceso de GitHub a Supabase	20
Ilustración 22 Crear una nueva organización en Supabase	21
Ilustración 23 Panel inicial del proyecto en Supabase.....	22
Ilustración 24 Sección de autenticación en Supabase	22
Ilustración 25 Configuración de inicio de sesión en Supabase.....	23
Ilustración 26 onfiguración general del proyecto en Supabase	24
Ilustración 27 Configuración de la API de datos en Supabase	24
Ilustración 28 Claves API en Supabase	25
Ilustración 29 Integración de Supabase en el código JavaScript.....	25
Ilustración 30 Usuario registrado en Supabase.....	26
Ilustración 31 usuarios registrados.....	26
Ilustración 32 Función para agregar estudiantes en Supabase.....	27
Ilustración 33 Formulario para registrar estudiantes en HTML	28
Ilustración 34 Crear nueva tabla en Supabase.....	29
Ilustración 35 Configuración inicial de la nueva tabla en Supabase	30
Ilustración 36 Asignación de nombre y configuración de seguridad en la tabla	30
Ilustración 37 Definición de columnas en la tabla estudiantes.....	31
Ilustración 38 Tabla estudiantes creada en Supabase	32
Ilustración 39 Acceso a políticas de seguridad en la tabla estudiantes	32
Ilustración 40 Crear una nueva política en Supabase	33
Ilustración 41 Configuración de política de lectura en Supabase	33
Ilustración 42 Política de lectura para usuario autenticado en Supabase.....	34
Ilustración 43 Selección de roles objetivo en política de Supabase.....	34
Ilustración 44 Política creada exitosamente en Supabase	34
Ilustración 45 Interfaz del Panel de Estudiantes.....	35
Ilustración 46 usuarios subidos a la base de datos	35
Ilustración 47 Acceder a la sección Storage en Supabase.....	36

Ilustración 48 Crear un bucket de almacenamiento	37
Ilustración 49 Acceder al bucket creado.....	37
Ilustración 50 Configurar políticas de almacenamiento	38
Ilustración 51 Agregar nueva política al bucket.....	39
Ilustración 52 Configurar permisos de la política.....	40
Ilustración 53 Definir roles y condición de acceso.....	40
Ilustración 54 Subir archivos desde la aplicación	41
Ilustración 55 Verificación de archivos en el almacenamiento.....	41

INTRODUCCIÓN

La presente práctica tiene como objetivo integrar herramientas y plataformas en la nube para el desarrollo, despliegue y administración de una aplicación web moderna. A través de este proceso se busca comprender y aplicar el ciclo completo de trabajo: desde la creación y manejo de repositorios en GitHub, la implementación continua (CI/CD) con Vercel, hasta la configuración de una base de datos y almacenamiento en Supabase. Con ello, se fomenta el aprendizaje de tecnologías que permiten el desarrollo colaborativo, la publicación rápida y la gestión segura de datos y archivos, optimizando el flujo de trabajo en entornos reales de producción.

REQUERIMIENTOS

Herramientas y servicios utilizados:

1. GitHub

- Creación de repositorio.
- Clonado y edición del proyecto desde GitHub Desktop y Visual Studio Code.
- Control de versiones mediante commits.

2. Vercel

- Integración con GitHub para despliegue automático.
- Configuración de proyecto y dominio.
- Verificación de estado de producción.

3. Supabase

- Creación de organización y proyecto.
- Configuración de autenticación de usuarios.
- Creación y gestión de tablas (Base de datos).
- Configuración de políticas de seguridad (RLS).
- Integración de claves API en la aplicación.
- Gestión de almacenamiento (buckets) y permisos de acceso.

Recursos técnicos necesarios:

- Computadora con conexión a Internet.
- Navegador web actualizado.
- Cuenta activa en GitHub, Vercel y Supabase.

- Editor de código (Visual Studio Code).
- GitHub Desktop instalado.

CREAR REPOSITORIO

Paso 1:

Creamos un nuevo repositorio. Agregamos el nombre del repositorio en el campo correspondiente (por ejemplo, clase-cloud), verificamos que esté disponible y, si es necesario, agregamos una descripción. Posteriormente, seleccionamos la visibilidad del repositorio (público o privado) para continuar con la configuración.

The screenshot shows the 'Create new repository' form on GitHub. It's a two-step process. Step 1: General is currently active. The 'Dueño' dropdown is set to 'DAVID08FR'. The 'Nombre del repositorio' input field contains 'clase-cloud', which is highlighted in green with the message 'clase-cloud está disponible.' Below it, a note says 'Los buenos nombres de repositorios son cortos y fáciles de recordar. ¿Qué tal? Eureka refactorizado?' The 'Descripción' field is empty with a placeholder '0 / 350 caracteres'. Step 2: Configuración is shown below with a visibility dropdown set to 'Público'. A note at the bottom says 'Elija quién puede ver y comprometerse con este repositorio'.

Ilustración 1 Creación de un nuevo repositorio en GitHub. Autoría Propia

Paso 2:

Después de crear el repositorio en GitHub, lo clonaremos en GitHub Desktop. Para ello, seleccionamos el repositorio clase-cloud de la lista, elegimos la carpeta de destino con la opción Choose..., verificamos la ruta en Local path y damos clic en Clone para descargarlo en nuestra computadora.

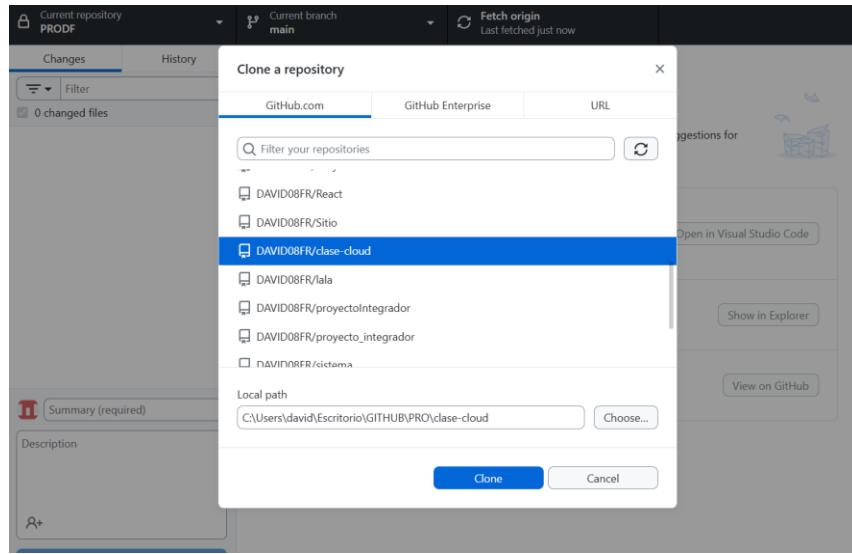


Ilustración 2 Clonar un repositorio desde GitHub Desktop. Autoría Propia

Paso 3:

Una vez clonado el repositorio, lo abrimos en Visual Studio Code. Para ello, damos clic en el botón Open in Visual Studio Code desde GitHub Desktop, lo que nos permitirá comenzar a editar y trabajar en el proyecto.

No local changes

There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.

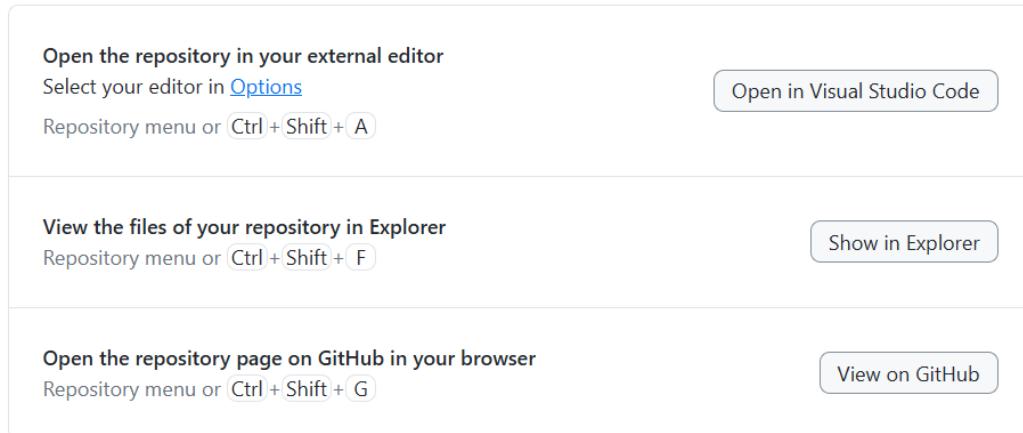


Ilustración 3 Abrir el repositorio en Visual Studio Code. Autoría Propia

Con el repositorio abierto en Visual Studio Code, verificamos que la carpeta del proyecto (clase-cloud) aparezca en el panel lateral izquierdo. Desde aquí podremos crear, editar y guardar los archivos que formarán parte del repositorio.

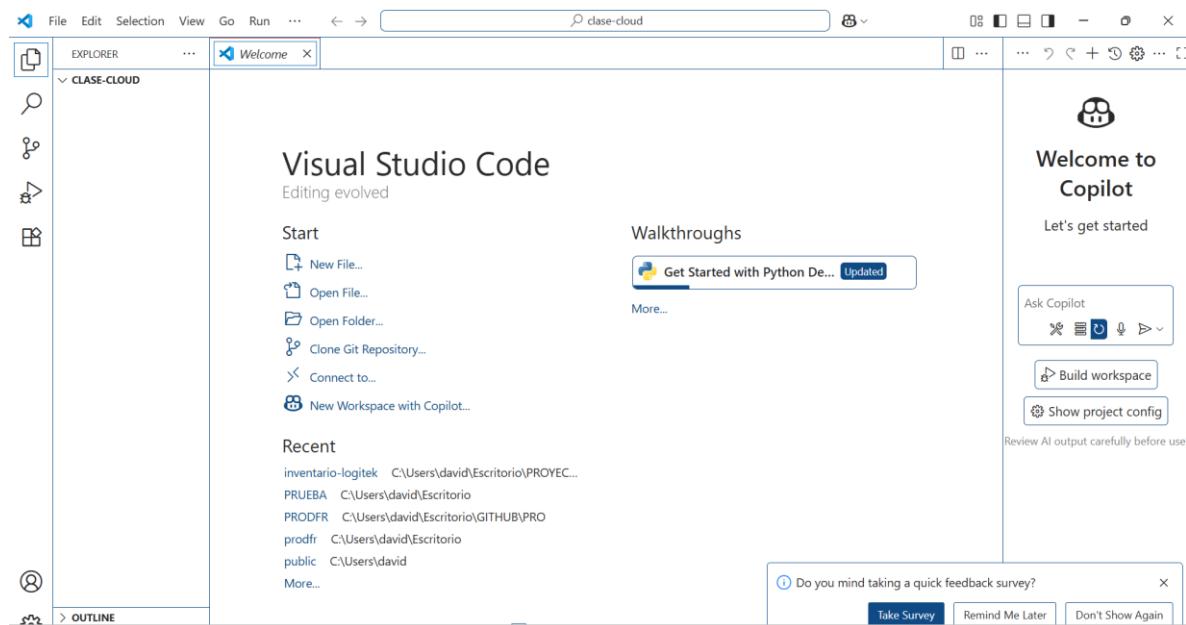


Ilustración 4 Visual Studio Code con el repositorio abierto. Autoría Propia

Agregamos la estructura base del proyecto, incluyendo el archivo index.html con las etiquetas principales (`<!DOCTYPE html>`, `<html>`, `<head>` y `<body>`). En este archivo definimos el título de la página, enlazamos la hoja de estilos styles.css y escribimos las primeras líneas de código HTML para la interfaz, como el formulario de inicio de sesión y registro.



```

CLASE... js dashboard.html index.html styles.css
index.html
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4   <meta charset="UTF-8" />
5   <title>Mi Clase en la Nube - Login</title>
6   <link rel="stylesheet" href="styles.css" />
7 </head>
8 <body>
9   <h1>Bienvenida a Mi Clase en la Nube</h1>
10
11  <div id="auth-container">
12
13    <div id="login-form">
14      <h2>Iniciar sesión</h2>
15      <input type="email" id="email" placeholder="correo" />
16      <input type="password" id="password" placeholder="contraseña" />
17      <button onclick="login()">Entrar</button>
18      <p>No tienes cuenta? <a href="#" onclick="toggleForms()">Regístrate aquí</a></p>
19    </div>
20
21    <div id="register-form" style="display:none;">
22      <h2>Registrarse</h2>
23      <input type="email" id="reg-email" placeholder="Correo" />
24      <input type="password" id="reg-password" placeholder="contraseña" />
25      <button onclick="register()">Crear cuenta</button>
26      <p>Ya tienes cuenta? <a href="#" onclick="toggleForms()">Inicia sesión aquí</a></p>
27    </div>
28
29  </div>
30
31  <script src="https://cdn.jsdelivr.net/npm/@supabase/supabase-js"></script>
32  <script src="js/auth.js"></script>

```

Ilustración 5 Creación de la estructura HTML del proyecto. Autoría Propia

Paso 6:

Después de agregar la estructura HTML y los estilos, regresamos a GitHub Desktop para guardar los cambios. Verificamos los archivos modificados (index.html y styles.css), escribimos un mensaje descriptivo en el campo Summary y damos clic en Commit to main para registrar los cambios en el repositorio local.

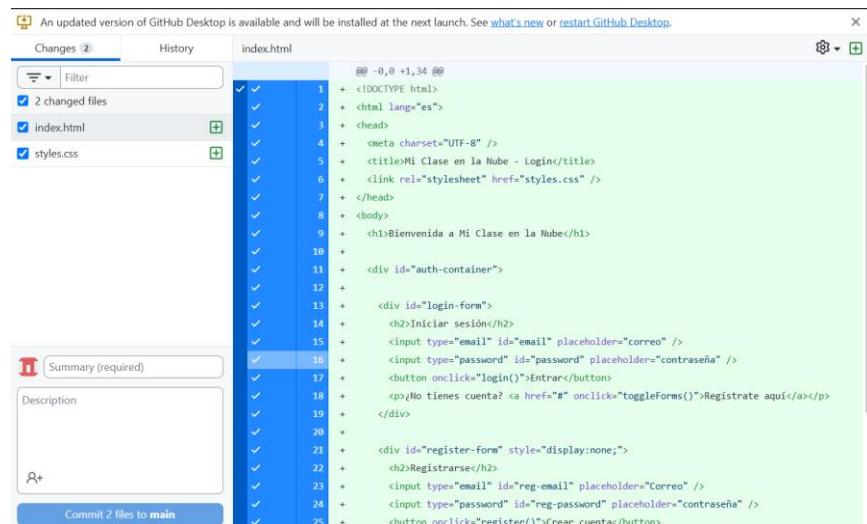


Ilustración 6 Guardar cambios y realizar commit en GitHub Desktop. Autoría Propia

INICIAR SESIÓN VERCEL

En nuestro navegador buscamos Vercel y accedemos a su página oficial. Al ingresar, seleccionamos la opción para iniciar sesión y elegimos el método de acceso que prefiramos, como Continuar con Google o Continuar con GitHub, para vincular nuestra cuenta y continuar con la configuración del despliegue.

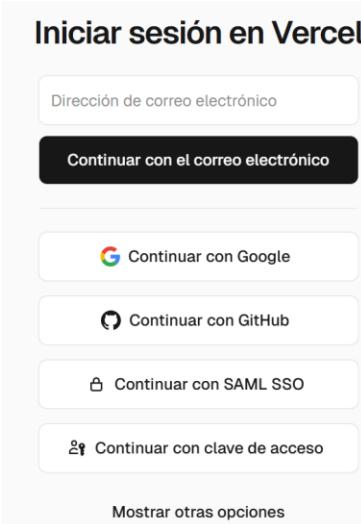


Ilustración 7 Iniciar sesión en Vercel. Autoría Propia

Paso 8:

Después de iniciar sesión con GitHub en Vercel, autorizamos el acceso para que la plataforma pueda conectarse a nuestros repositorios. Verificamos los permisos solicitados, como la lectura de direcciones de correo electrónico y acceso a repositorios, y damos clic en Autorizar Vercel para continuar con la integración.

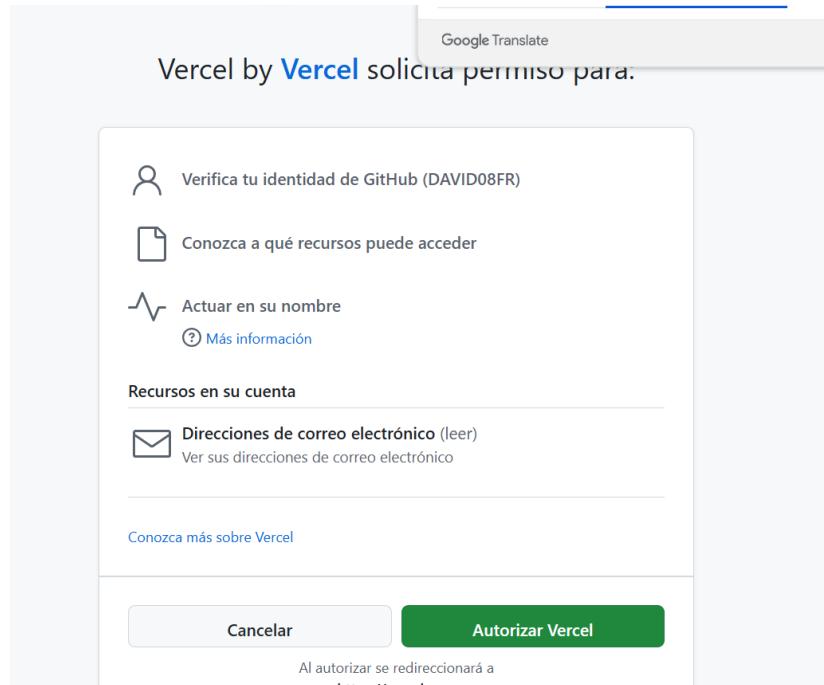


Ilustración 8 Autorizar permisos de GitHub en Vercel. Autoría Propia

Paso 9:

Una vez autorizada la cuenta, accedemos al panel principal de Vercel. Desde aquí podremos implementar nuestro primer proyecto seleccionando la opción Importar para vincular un repositorio de GitHub o usar una plantilla predeterminada para comenzar el despliegue.

A screenshot of the Vercel dashboard. The top navigation bar includes "Proyectos de davidfr339-gmail...", "Pasatiempo", "Encontrar...", "Comentario", "Añadir n...", and tabs for "Descripción general", "Integraciones", "Despliegues", "Actividad", "Dominios", "Uso", "Observabilidad", "Almacenamiento", "Banderas", "Puerta de enlace de IA", "Apoyo", and "Ajustes". The main area has a search bar "Buscar Proyectos...". On the left, there's a "Uso" section showing "Últimos 30 días" (Actualizado hace 4m) with metrics for data transfer and edge requests, and an "Avances recientes" section. On the right, there's a "Proyectos" section with a large "Subir" button, a "Implementa tu primer proyecto" section with "Proyecto de importación" and "Plantilla repetitiva de Next.js" options, and a "Importar" and "Desplegar" button.

Ilustración 9 Panel principal de Vercel. Autoría Propia

IMPORTAR PROYECTO DE GITHUB A VERCEL

Paso 10:

Para comenzar el despliegue, seleccionamos la opción Proyecto de importación y damos clic en Importar. Esto nos permitirá elegir un repositorio desde nuestro proveedor Git (por ejemplo, GitHub) y vincularlo con Vercel para su implementación en línea.

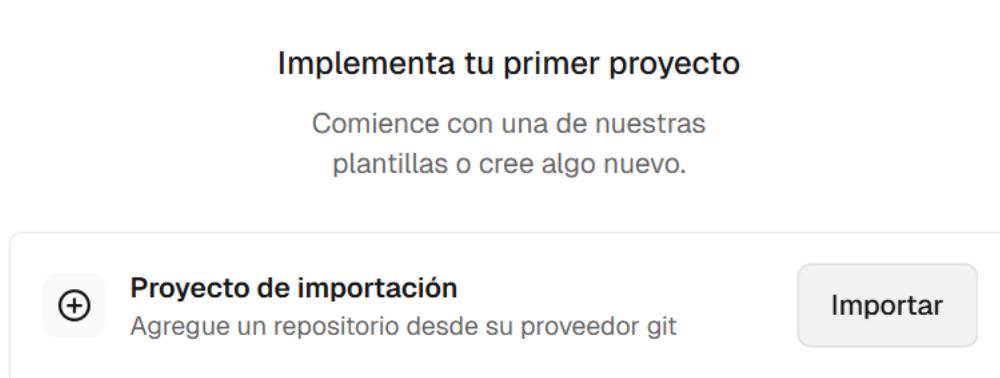


Ilustración 10 Importar un proyecto en Vercel. Autoría Propia

Paso 11:

En la pantalla de Importar repositorio Git, seleccionamos el proveedor donde tenemos alojado nuestro proyecto. En este caso, damos clic en Continuar con GitHub para conectar Vercel con nuestro repositorio y proceder con la importación.

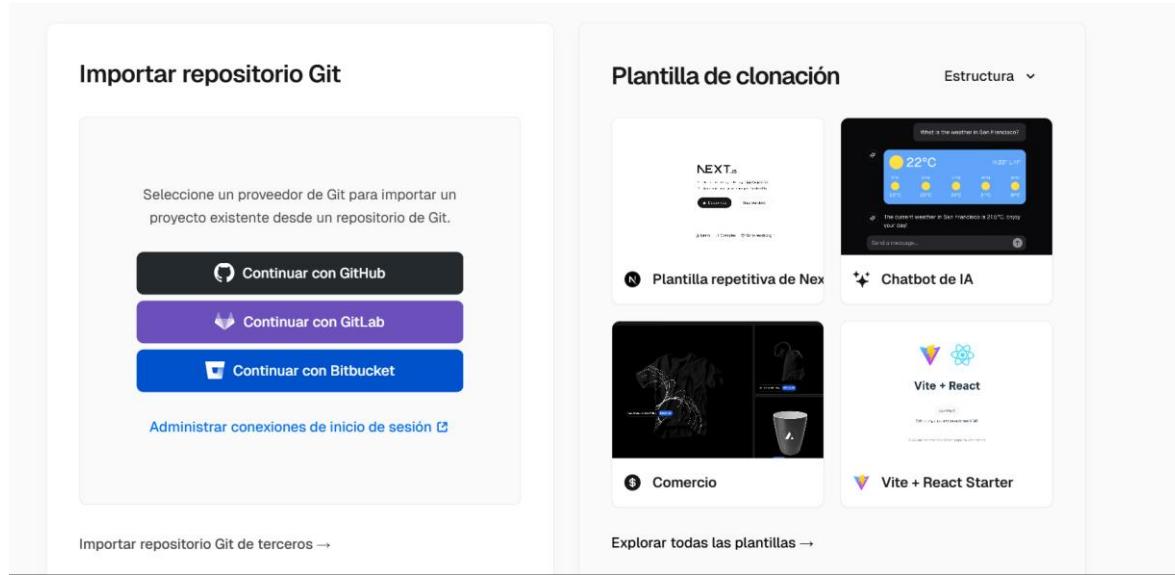


Ilustración 11 Seleccionar proveedor para importar repositorio. Autoría Propia

Paso 12:

Seleccionamos si queremos dar acceso a todos los repositorios o solo a repositorios específicos. En este caso, marcamos Todos los repositorios para que Vercel pueda acceder a todos los proyectos actuales y futuros en GitHub. Confirmamos los permisos de lectura y escritura y continuamos con la instalación.

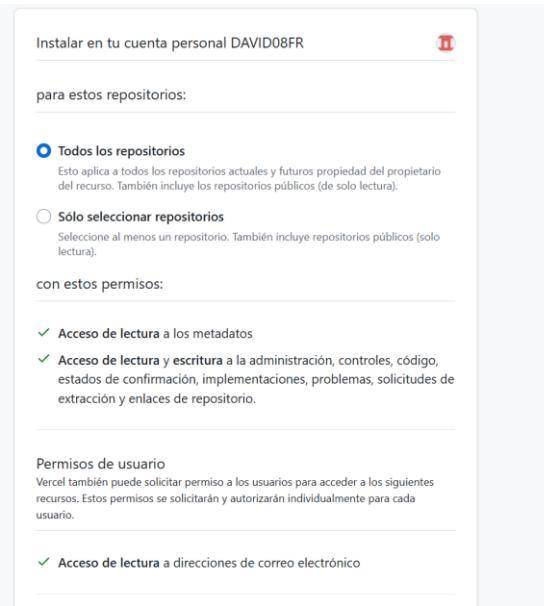


Ilustración 12 Conceder permisos de acceso a repositorios en Vercel. Autoría Propia

Paso 13:

Ingresamos nuestra contraseña de GitHub para confirmar el acceso en modo seguro (modo sudo). Este paso es necesario para autorizar cambios o integraciones importantes, como la conexión de repositorios con Vercel. Luego, damos clic en Confirmar para continuar.



Ilustración 13 Confirmar acceso en GitHub. Autoría Propia

Paso 14:

En la lista de repositorios disponibles, buscamos y seleccionamos clase-cloud. Luego, damos clic en Importar para comenzar el proceso de implementación de este proyecto en Vercel.

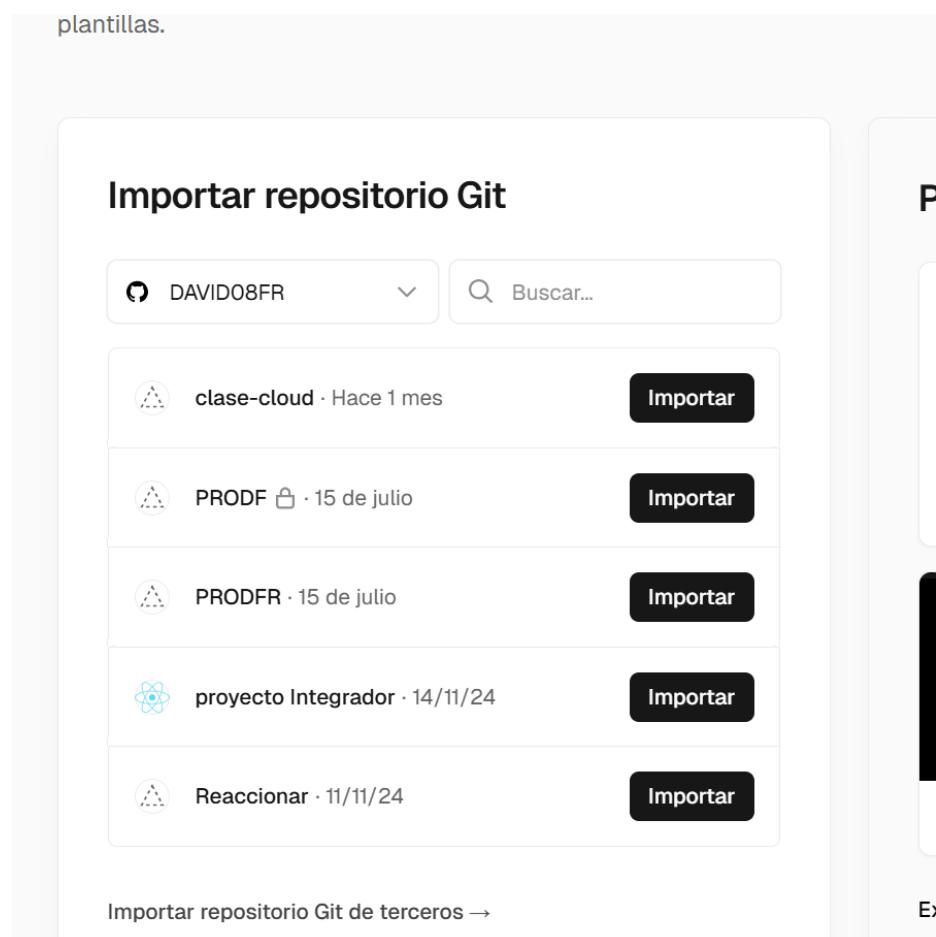


Ilustración 14 Seleccionar repositorio para importar en Vercel. Autoría Propia

Paso 15:

Asignamos el nombre del proyecto (en este caso, clase-cloud), verificamos el marco preestablecido como Other y confirmamos el directorio raíz del proyecto. Una vez revisados estos datos, damos clic en Desplegar para iniciar la publicación del sitio en Vercel.

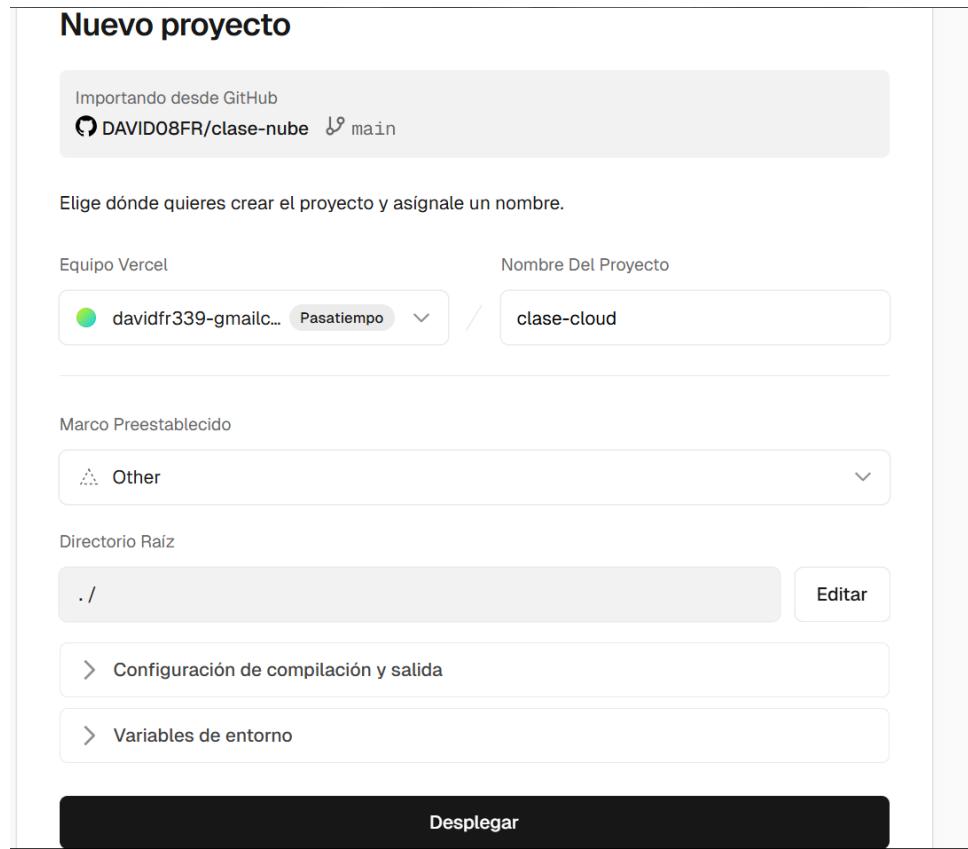


Ilustración 15 Configuración del nuevo proyecto en Vercel. Autoría Propia

Paso 16:

¡Listo! El proyecto clase-cloud se ha implementado correctamente en Vercel. Ahora podemos visualizarlo en línea y compartir el enlace generado para que otros puedan acceder a la aplicación desde cualquier navegador.

¡Felicidades!

Acabas de implementar un nuevo proyecto para  Proyectos de davidfr339-gmailcom .



Próximos pasos

Ilustración 16 Proyecto implementado con éxito en Vercel. Autoría Propia

Paso 17:

En el panel de Despliegue de producción de Vercel, verificamos que el estado aparezca como Listo. Aquí podemos ver el dominio asignado (clase-cloud.vercel.app), la rama utilizada (main) y el historial de implementaciones. Desde esta vista también podemos acceder a la configuración y registros de compilación.

The screenshot shows the Vercel dashboard for a deployment. On the left, there's a preview window titled "Bienvenida a Mi Clase en la Nube" with fields for "correo" and "contraseña" and a "Entrar" button. To the right, the deployment details are listed:

- Despliegue:** clase-cloud-bmn7x4efh-davidfr339-gmailcoms-proyectos.vercel.app
- Dominios:** clase-cloud.vercel.app +2
- Estado:** Creado
- Lista:** Hace 54 s por DAVID08FR
- Fuente:** main
- 86d2234 confirmación de primer paso

At the bottom, there are configuration options like "Configuración de implementación", "Cálculo de fluidos", "Protección de implementación", and "Protección contra sesgos". A note says "Para actualizar su implementación de producción, envíala a la rama principal." and a "Despliegos" button.

Ilustración 17 Vista del despliegue de producción en Vercel. Autoría Propia

Paso 18:

Abrimos el dominio generado por Vercel (clase-cloud.vercel.app) en el navegador y comprobamos que la aplicación se carga correctamente, mostrando la página de inicio de sesión tal como fue diseñada en el proyecto.

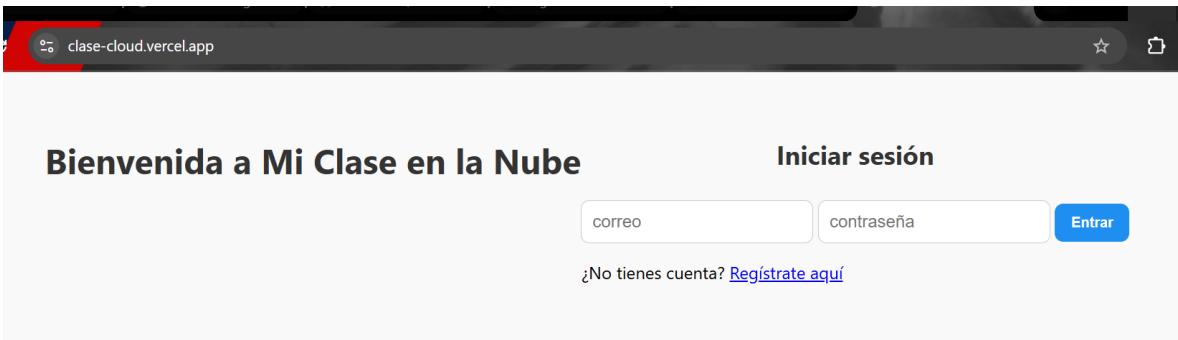


Ilustración 18 Proyecto funcionando en dominio de Vercel. Autoría Propia

INICIAR SESIÓN EN SUPABASE Y VINCULAR CON GITHUB

Paso 19:

En el navegador, buscamos Supabase y accedemos a su página oficial. Desde aquí podremos crear un nuevo proyecto para configurar nuestra base de datos y funcionalidades de autenticación que utilizaremos en nuestra aplicación.



Ilustración 19 Página principal de Supabase. Autoría Propia

Paso 20:

En la pantalla de inicio de sesión de Supabase, elegimos el método de acceso. Podemos usar Continuar con GitHub para vincular directamente nuestra cuenta o ingresar manualmente el correo electrónico y contraseña para iniciar sesión y acceder al panel de proyectos.



Ilustración 20 Iniciar sesión en Supabase. Autoría Propia

Paso 21:

Al iniciar sesión con GitHub en Supabase, se nos solicita autorizar el acceso a nuestra cuenta. Revisamos los permisos solicitados (como lectura de direcciones de correo electrónico) y damos clic en Autorizar supabase para completar la vinculación.

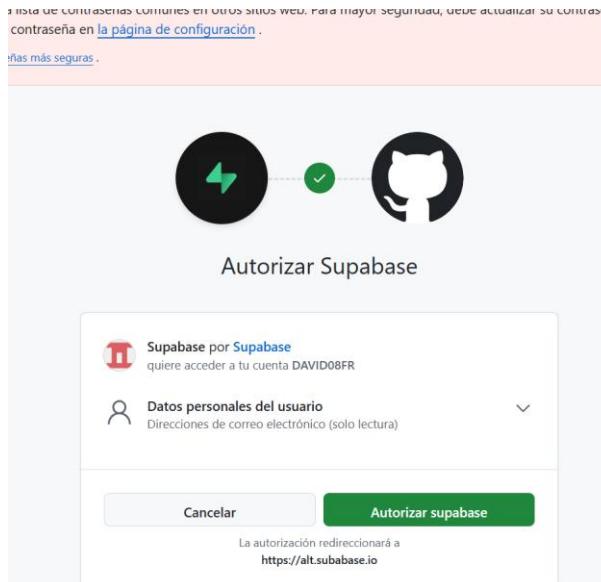


Ilustración 21 Autorizar acceso de GitHub a Supabase. Autoría Propia

Paso 22:

En la pantalla de creación de organización, asignamos un nombre (por ejemplo, DAVID08FR's Org), seleccionamos el tipo como Personal y elegimos el plan Free (\$0/month). Finalmente, damos clic en Crear organización para continuar con la configuración del proyecto. Barcelona@12

Crear una nueva organización

Esta es su organización dentro de Supabase.
Por ejemplo, puede utilizar el nombre de su empresa o departamento.

Nombre
DAVID08FR's Org
¿Cómo se llama tu empresa o equipo?

Tipo
Personal
¿Qué describiría mejor su organización?

Plan
Free - \$0/month
El Plan se aplica a su nueva organización.

Puedes cambiar el nombre de tu organización más tarde

Cancelar Crear organización

Ilustración 22 Crear una nueva organización en Supabase. Autoría Propia

Paso 23:

Accedemos al panel del nuevo proyecto en Supabase, donde podemos comenzar a construir la base de datos. Desde aquí, tenemos acceso al Editor de tablas, Editor de SQL y la configuración de la base de datos para crear tablas, insertar datos y gestionar nuestra API.

Proyecto de DAVID08FR NANO

Tablas Funciones Rélicas 0 0 0 Estado del proyecto

Bienvenido a tu nuevo proyecto

Su proyecto se ha implementado en su propia instancia, con su propia API configurada y lista para usar.

Comience a construir su base de datos

Empieza a crear tu aplicación creando tablas e insertando datos. Nuestro Editor de Tablas hace que Postgres sea tan fácil de usar como una hoja de cálculo, pero también puedes usar nuestro Editor SQL si necesitas algo más.

Editor de tablas Editor de SQL Acerca de la base de datos

	identificación	estado
1	Crear un proyecto	Completo
2	Leer la documentación	Completo
3	Crear aplicación	En curso
4	Conectar Supabase	En curso
5	Implementar proyecto	No iniciado
6	Conseguir usuarios	No iniciado
7	Actualizar a Beta	No iniciado

Ilustración 23 Panel inicial del proyecto en Supabase. Autoría Propia

AUTENTICACION.

Paso 24:

Dentro del panel del proyecto en Supabase, ingresamos a la sección Autenticación para gestionar el registro e inicio de sesión de usuarios. Desde aquí podemos agregar usuarios manualmente o configurar el sistema para que los usuarios se registren directamente desde la aplicación.

Organización de DAVID08FR Gratis / Proyecto de DAVID08FR principal Producción Conectar Comentario

Descripción general... Editor de tablas Editor de SQL Base de datos Autenticación Almacenamiento Funciones de borde Tiempo real Asesores Informes Registros Documentación de ...

Usuarios

Buscar correo electrónico, teléfono Todos los usuarios Proveedor Todas las columnas Refrescar Agregar usuario

Ordenado por creado en

UID	Nombre para mostrar	Correo electrónico	Teléfono
No hay usuarios en tu proyecto. Actualmente no hay usuarios registrados en tu proyecto.			

Ilustración 24 Sección de autenticación en Supabase. Autoría Propia

Paso 25:

En la sección Iniciar sesión / Proveedores de Supabase, configuramos las opciones de registro e inicio de sesión. Activamos Permitir que nuevos usuarios se registren y Confirmar correo electrónico para validar las cuentas antes del primer acceso. Estas opciones garantizan mayor seguridad y control en el sistema de autenticación.

The screenshot shows the Supabase dashboard with the URL <https://supabase.com/dashboard/project/1zbheusaxnenhwadvn/auth/providers>. The left sidebar is titled 'Autenticación' and includes sections for 'ADMINISTRAR' (Users), 'CONFIGURACIÓN' (Configuration), and 'Iniciar sesión / Proveedores' (Session Providers). The 'Iniciar sesión / Proveedores' section is selected and expanded, showing sub-options like 'Sesiones', 'Límites de velocidad', 'Correos electrónicos', 'Multifactorial', 'Configuración de URL', 'Protección contra ataques', 'Ganchos de autenticación... (BETA)', and 'Avanzado'. The main content area is titled 'Iniciar sesión / Proveedores' and describes configuring providers for authentication and session methods. It has two tabs: 'Autenticación de Supabase' (selected) and 'Autorización de terceros'. Below this, the 'Registros de usuarios' (User Registrations) section contains four toggle switches: 'Permitir que nuevos usuarios se registren' (Allow new users to register) is turned on (green), 'Permitir la vinculación manual' (Allow manual linking) is turned off (grey), 'Permitir inicios de sesión anónimos' (Allow anonymous sign-ins) is turned off (grey), and 'Confirmar correo electrónico' (Verify email address) is turned on (green). A note below the first option states: 'Si esta opción está deshabilitada, los nuevos usuarios no podrán registrarse en su aplicación.'

Ilustración 25 Configuración de inicio de sesión en Supabase. Autoría Propia

Paso 26:

En la sección Configuración del proyecto, podemos cambiar el nombre del proyecto, copiar la identificación del proyecto para usarla en nuestra aplicación y acceder a opciones como reiniciar o pausar el proyecto. Estos ajustes permiten administrar el entorno de producción de manera flexible.

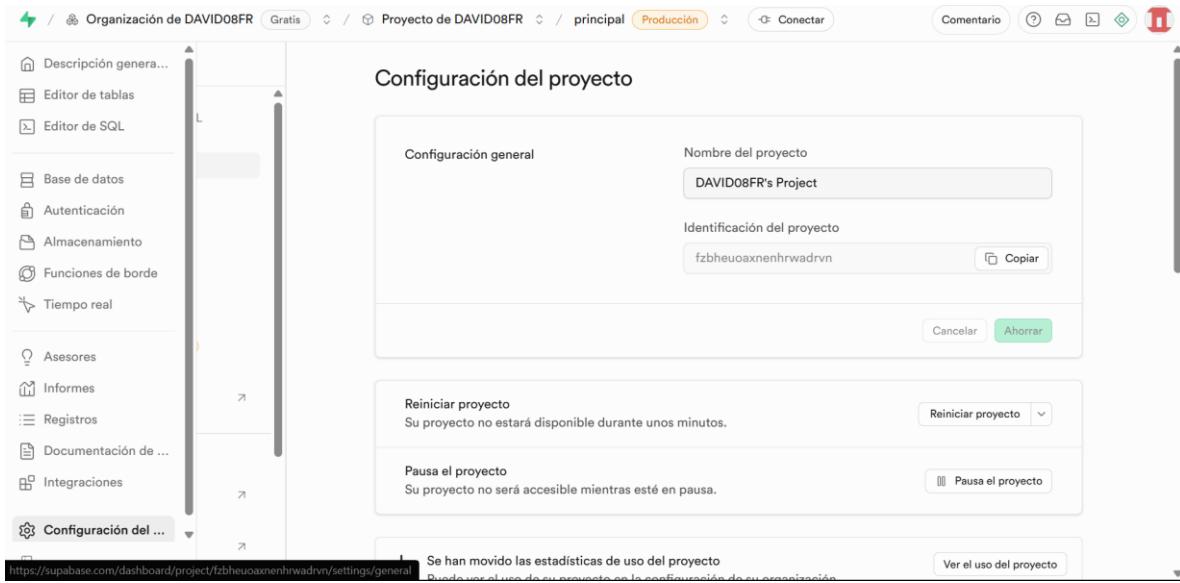


Ilustración 26 onfiguración general del proyecto en Supabase. Autoría Propia

Paso 27:

En la sección API de datos, activamos la opción Habilitar API de datos para permitir el uso de las bibliotecas cliente de Supabase y endpoints PostgREST. También verificamos los esquemas expuestos (por ejemplo, public) que estarán disponibles para consultas y administración desde nuestra aplicación.

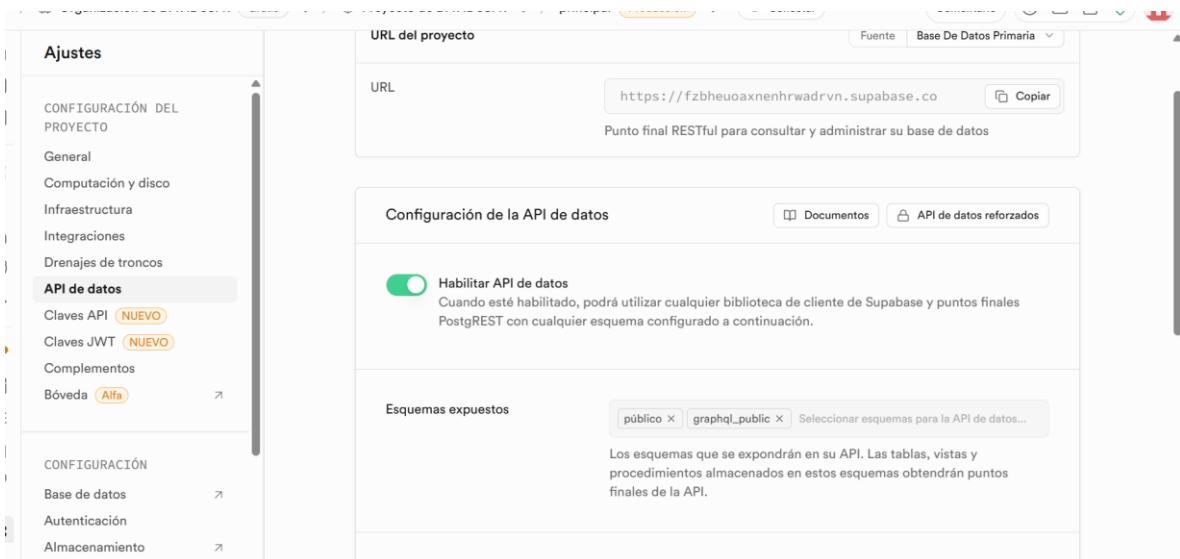


Ilustración 27 Configuración de la API de datos en Supabase. Autoría Propia

<https://fzbheuoaxnenhrwadrvn.supabase.co>

Paso 28:

En la sección Claves API, obtenemos las credenciales necesarias para conectar nuestra aplicación con la base de datos de Supabase. La clave anon (public) se utiliza para operaciones seguras desde el cliente, mientras que la clave service_role (secret) se reserva para tareas administrativas y nunca debe compartirse públicamente.

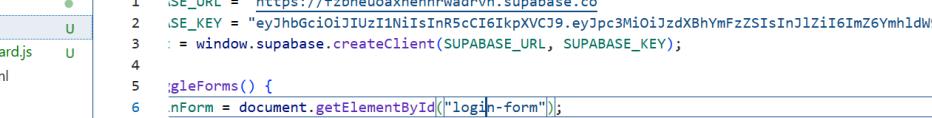


Ilustración 28 Claves API en Supabase. Autoría Propia

eyJhbGciOiJIUzI1NilsInR5cCl6IkpxVCJ9.eyJpc3MiOiJzdXBhYmFzZSIslnJlZl6ImZ6YmhldW9heG5lbmhvd2FkcnZuliwicm9sZSI6ImFub24iLCJpYXQiOjE3NTQ1MDUwOTMsImV4CI6MjA3MDA4MTA5M30.49jF_IMBuvKt-HWpmN-9USgw6tDqUKzrW1uaOqNFi40

Paso 29:

En el archivo auth.js, agregamos las credenciales de Supabase (SUPABASE_URL y SUPABASE_KEY) para inicializar la conexión con la base de datos. También implementamos funciones como toggleForms() para alternar entre el formulario de inicio de sesión y el de registro, y onRegister() para registrar nuevos usuarios mediante la API de autenticación de Supabase.



The screenshot shows the VS Code interface with the following details:

- Explorer View:** Shows the project structure under "CLASE-CLOUD". The "auth.js" file is currently selected.
- Editor View:** Displays the content of the "auth.js" file. The code handles form toggling and user sign-up using Supabase's JavaScript client library.
- Status Bar:** Shows the file path as "CLASE-CLOUD/auth.js" and the status "1 file(s) changed".

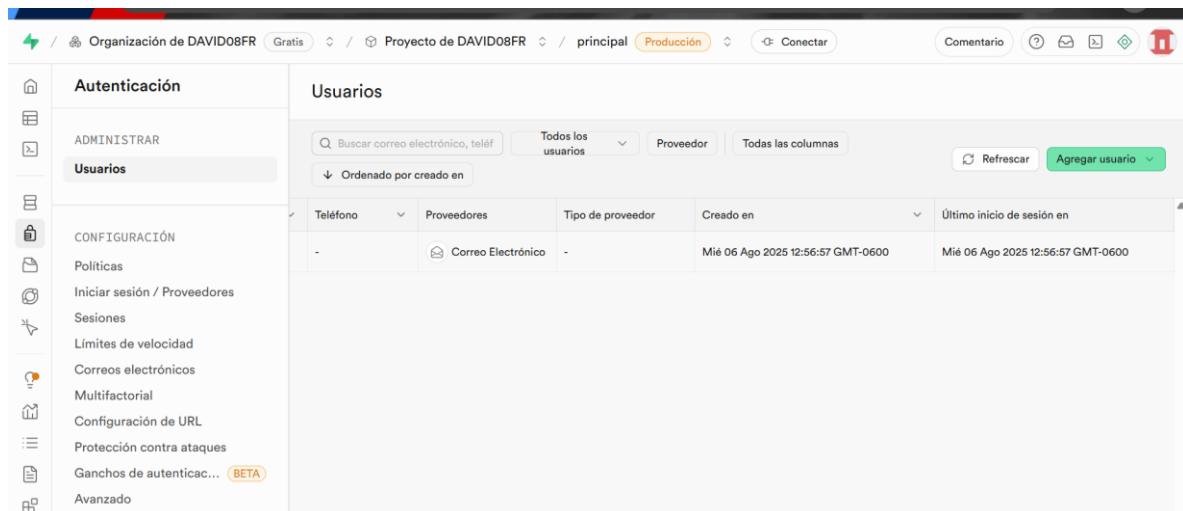
```
CLASE-CLOUD
  js
    auth.js
    dashboard.js
  index.html
  styles.css

js > auth.js auth.js > toggleForms @ loginForm
1  :SE_URL = "https://fbzheuoaxnenhrwadrn.supabase.co"
2  :SE_KEY = "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJp3M0iJzdXBhYmFzZSIsInJlZiI6ImZ6Ymh1dW9h
3  := window.supabase.createClient(SUPABASE_URL, SUPABASE_KEY);
4
5  gleForms() {
6    .nForm = document.getElementById("login-form");
7    .sterForm = document.getElementById("register-form");
8
9    style.display = loginForm.style.display === "none" ? "block" : "none";
10   rrm.style.display = registerForm.style.display === "none" ? "block" : "none";
11
12
13  .on register() {
14    .1 = document.getElementById("reg-email").value;
15    .word = document.getElementById("reg-password").value;
16
17    ita, error } = await client.auth.signUp({
18      ...
19    },
```

Ilustración 29 Integración de Supabase en el código JavaScript. Autoría Propia

Paso 30:

En la sección Usuarios de Supabase, verificamos que el registro desde la aplicación se haya realizado correctamente. Aquí podemos ver el correo electrónico del usuario, el tipo de proveedor utilizado para el inicio de sesión y la fecha de creación de la cuenta.



The screenshot shows the Supabase dashboard with the URL [https://supabase-organization-12345.supabase.co/auth/users](#). The left sidebar has sections for Autenticación, Administrar, and Usuarios. The Usuarios section is selected. The main area is titled "Usuarios" with a search bar and filters for "Todos los usuarios", "Proveedor", and "Todas las columnas". A table lists one user: "Correo Electrónico" (pablo@gmail.com) with "Teléfono" (empty), "Tipo de proveedor" (empty), "Creado en" (Mié 06 Ago 2025 12:56:57 GMT-0600), and "Último inicio de sesión en" (Mié 06 Ago 2025 12:56:57 GMT-0600). There are buttons for "Refrescar" and "Agregar usuario".

Ilustración 30 Usuario registrado en Supabase. Autoría Propia



The screenshot shows the Supabase dashboard with the URL [https://supabase-organization-12345.supabase.co/auth/users](#). The left sidebar has sections for Autenticación, Administrar, and Usuarios. The Usuarios section is selected. The main area is titled "Usuarios" with a search bar and filters for "Todos los usuarios", "Proveedor", and "Todas las columnas". A table lists two users: "pablo@gmail.com" and "david@gmail.com", both with empty "Nombre para mostrar", "Teléfono", and "P". There are buttons for "Refrescar" and "Agregar usuario".

Ilustración 31 usuarios registrados. Autoría Propia

Paso 31:

En el archivo `dashboard.js`, creamos una función para agregar estudiantes a la base de datos de Supabase. Esta función obtiene los valores de los campos del formulario, verifica si el usuario está autenticado y, en caso afirmativo, inserta un nuevo registro en la tabla estudiantes. Si ocurre un error, muestra un mensaje de alerta; de lo contrario, confirma que el estudiante fue agregado exitosamente.

The screenshot shows a code editor interface with a sidebar on the left displaying a file tree. The tree includes a folder named 'CLASE-CLOUD' containing 'js' (which further contains 'auth.js', 'dashboard.js' (selected), and 'index.html'), and '# styles.css'. Below the tree is a button labeled '> OUTLINE'. The main area of the editor displays the contents of 'dashboard.js'. The code uses Supabase's JavaScript client to interact with a database. A tooltip for the 'getElementById' method is shown, stating: '(method) Document.getElementById(elementId: string): HTMLElement | null Returns a reference to the first object with the specified value of the ID attribute. @param elementId — String that specifies the ID value.' The code itself includes logic for user authentication and inserting new student records into a 'estudiantes' table.

```
const SUPABASE_URL = "https://fzbheuoaxnenhrwadrn.supabase.co";
const SUPABASE_KEY = "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJzdXBhYmFzZSIsInJlZii6I";
// Crear cliente de Supabase
const client = supabase.createClient(SUPABASE_URL, SUPABASE_KEY);

// Función para agregar un estudiante
async function agregarEstudiante() {
  const nombre = document.getElementById("nombre").value;
  const correo = document.getElementById("correo").value;
  const clase = document.getElementById("clase").value;

  const { data: { user }, error: userError } = await client.auth.getUser();

  if (userError || !user) {
    alert("No estás autenticado.");
    return;
  }

  const { error } = await client.from("estudiantes").insert({
    nombre,
    correo,
    clase,
    user_id: user.id,
  });

  if (error) {
    alert("Error al agregar: " + error.message);
  } else {
    alert("Estudiante agregado");
    cargarEstudiantes();
  }
}
```

Ilustración 32 Función para agregar estudiantes en Supabase. Autoría Propia

Paso 32:

En el archivo dashboard.html, creamos un formulario para registrar estudiantes con campos de nombre, correo y clase. Al presionar el botón Agregar, se ejecuta la función agregarEstudiante() definida en dashboard.js, que envía los datos a la base de datos en Supabase.

The screenshot shows a code editor interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, ...
- Toolbar:** Includes icons for back, forward, search, and file operations.
- Explorer:** Shows a tree view of files: CLASE..., js (auth.js, dashboard.js), index.html, and # styles.css.
- Code Editor:** Displays the HTML code for "dashboard.html". The code includes sections for student registration, a student list, and file upload.
- Bottom Status Bar:** Ln 21, Col 15, Spaces: 4, UTF-8, CRLF, {HTML}

```
<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8">
<title>Panel - Estudiantes</title>
<link rel="stylesheet" href="styles.css" />
<script src="https://cdn.jsdelivr.net/npm/@supabase/supabase-js@2"></script>
</head>
<body>
<div class="container">
<h1>Panel de Estudiantes</h1>
<button onclick="cerrarSesion()" class="cerrar-btn">Cerrar sesión</button>
<section class="card">
<h2>Registrar estudiante</h2>
<input type="text" id="nombre" placeholder="Nombre del estudiante" />
<input type="email" id="correo" placeholder="Correo" />
<input type="text" id="clase" placeholder="Clase" />
<button onclick="agregarEstudiante()">Agregar</button>
</section>
<section class="card">
<h2>Lista de estudiantes</h2>
<ul id="lista-estudiantes"></ul>
</section>
<section class="card">
<h2>Subir archivo</h2>
<label for="estudiante">Selecciona un estudiante:</label>
<select id="estudiante"></select><br /><br />
```

Ilustración 33 Formulario para registrar estudiantes en HTML. Autoría Propia

BASES DE DATOS

Paso 33:

En la sección Base de datos de Supabase, seleccionamos Crear una nueva tabla para almacenar la información de los estudiantes. Esta tabla contendrá las columnas necesarias como nombre, correo, clase y el ID del usuario que registró al estudiante.

The screenshot shows the Supabase dashboard interface. On the left, there's a sidebar with various project settings like 'Descripción general...', 'Editor de tablas', 'Editor de SQL', and 'Base de datos' (which is currently selected). The main area shows a schema named 'public' with the message 'Aún no se han creado tablas' (No tables have been created yet) and a button 'Crear una nueva tabla'. The URL at the bottom is <https://supabase.com/dashboard/project/7bhheusmnlwywadyn/databases/schemas>.

Ilustración 34 Crear nueva tabla en Supabase. Autoría Propia

Paso 34:

En el Editor de tablas, ingresamos el nombre de la tabla (por ejemplo, estudiantes) y, opcionalmente, una descripción. Dejamos habilitada la seguridad a nivel de fila (RLS) para proteger los datos, y luego continuamos agregando las columnas necesarias antes de guardar la tabla.

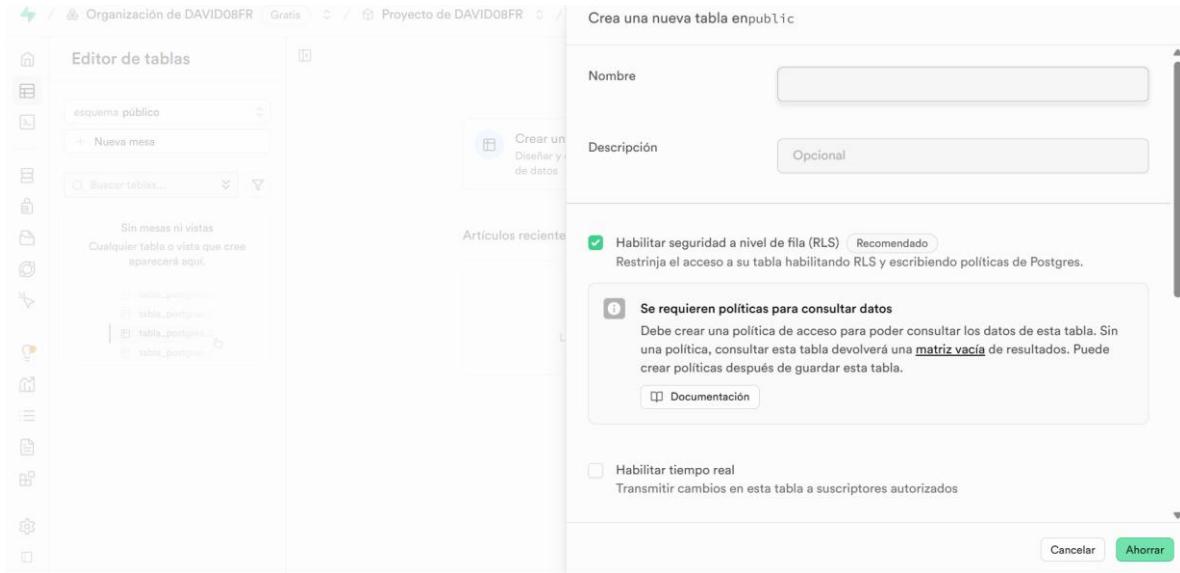


Ilustración 35 Configuración inicial de la nueva tabla en Supabase. Autoría Propia

Paso 35:

Asignamos el nombre estudiantes a la nueva tabla, mantenemos activa la opción Habilitar seguridad a nivel de fila (RLS) para restringir el acceso, y dejamos la descripción como opcional. Luego damos clic en Ahorrar para guardar la configuración inicial de la tabla.



Ilustración 36 Asignación de nombre y configuración de seguridad en la tabla. Autoría Propia

Agregamos las columnas necesarias para la tabla **estudiantes**:

- **id** (entero 8, clave primaria, valor predeterminado generado aleatoriamente)
- **created_at** (marca de tiempo, valor predeterminado now())
- **nombre** (texto)
- **correo** (texto)
- **user_id** (UUID, valor predeterminado generado aleatoriamente)

Una vez configuradas las columnas, guardamos la tabla para comenzar a almacenar registros.

The screenshot shows the Supabase Table Editor interface. At the top, there are three buttons: 'Columnas' (selected), 'Acerca de los tipos de datos', and 'Importar datos desde CSV'. Below this is a table with five rows, each representing a column definition. The columns are labeled 'Nombre', 'Tipo', 'Valor predeterminado', and 'Primario'. The rows are as follows:

Nombre	Tipo	Valor predeterminado	Primario
id	íntero 8	gen_random_	<input checked="" type="checkbox"/>
created_at	marcas d	now()	<input type="checkbox"/>
nombre	texto	NULO	<input checked="" type="checkbox"/>
correo	texto	NULO	<input checked="" type="checkbox"/>
user_id	UUID	gen_random_	<input checked="" type="checkbox"/>

At the bottom of the table area is a button labeled 'Agregar columna'.

Ilustración 37 Definición de columnas en la tabla estudiantes. Autoría Propia

Paso 37:

La tabla estudiante se ha creado correctamente en Supabase con todas las columnas definidas. Desde el Editor de tablas, ahora podemos insertar nuevos registros manualmente o mediante nuestra aplicación conectada a la base de datos.

The screenshot shows the Supabase Table Editor interface. On the left, there's a sidebar titled 'Editor de tablas' with a dropdown set to 'esquema público' and a button '+ Nueva mesa'. Below it is a search bar 'Buscar tablas...'. In the center, the 'estudiantes' table is displayed with the following columns: 'identificación' (type: UUID), 'creado...' (type: date), 'marcas de tie...', 'nombre' (type: text), and 'correo' (type: varchar). A message at the top right says 'Table estudiantes is good to go!'. Below the table, a message states 'Esta mesa está vacía' (This table is empty) and a button 'Importar datos desde CSV' (Import data from CSV).

Ilustración 38 Tabla estudiantes creada en Supabase. Autoría Propia

Paso 38:

En el menú de opciones de la tabla estudiantes, seleccionamos Ver políticas para gestionar las reglas de acceso y permisos. Aquí podremos definir qué usuarios o roles pueden consultar, insertar, actualizar o eliminar datos en la tabla.

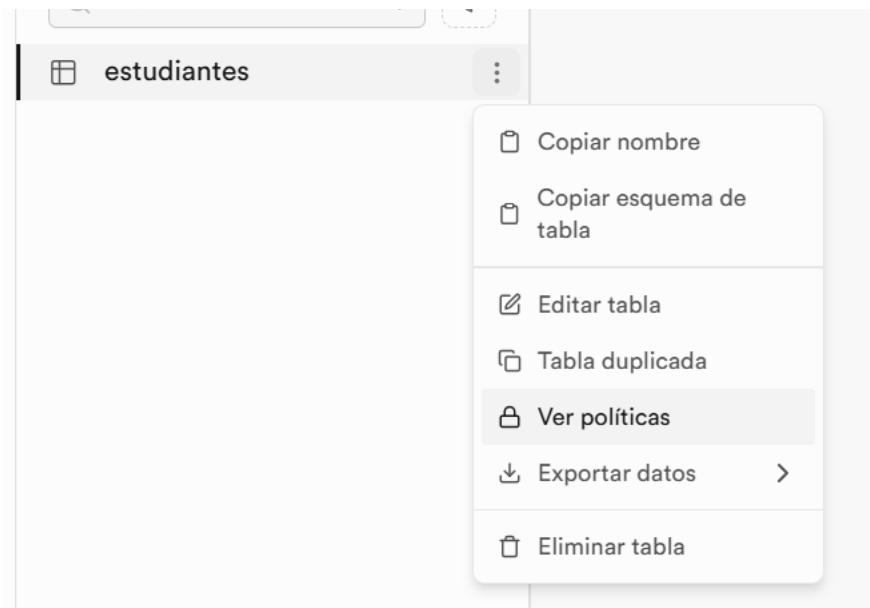


Ilustración 39 Acceso a políticas de seguridad en la tabla estudiantes. Autoría Propia

Paso 39:

Dentro de la configuración de políticas de la tabla, damos clic en Crear política para establecer reglas de acceso que controlen quién puede leer, insertar, actualizar o eliminar registros en la tabla estudiantes.

[Crear política](#)

Ilustración 40 Crear una nueva política en Supabase. Autoría Propia

Paso 40:

Creamos una política para seleccionar en la tabla estudiantes, con comportamiento Permisivo y comando de política SELECCIONAR. Esto permite habilitar el acceso de lectura a todos los usuarios, facilitando la visualización de los registros desde la aplicación.

The screenshot shows the Supabase policy creation interface. On the left, the configuration for a policy named "Politica para seleccionar" is shown. It specifies the table "estudiantes" and the condition "público . estudiante". The "Comportamiento político" is set to "Permisivo". The "Comando de política" is set to "SELECCIONAR". The "Roles objetivo" dropdown shows "ninguno" selected. Below this, there is a code editor with the following SQL query:

```
1  crear política " nombre_política "
2  en " público ". " estudiantes "
3  como PERMISIVO
4  para SELECCIONAR
5  al público
6  usando (
7  -- Proporcione una expresión SQL para la declaración de uso
8  ) ;
```

On the right, three policy options are listed:

- SELECCIONAR** **Habilitar acceso de lectura para todos los usuarios**
Esta política otorga acceso de lectura a su tabla a todos los usuarios a través de la operación SELECT.
- INSERTAR** **Habilitar inserción solo para usuarios autenticados**
Esta política otorga acceso de inserción a su tabla únicamente a todos los usuarios autenticados.
- ACTUALIZAR** **Habilitar actualizaciones para usuarios basadas en correo electrónico**
Esta política asume que su tabla tiene una columna "correo electrónico" y permite a los usuarios actualizar las filas cuya columna "correo electrónico" coincide con su correo electrónico.

Ilustración 41 Configuración de política de lectura en Supabase. Autoría Propia

Paso 41:

Definimos una política en la tabla estudiantes que permite seleccionar únicamente los registros cuyo id_usuario coincida con el identificador del usuario autenticado (auth.uid()). Esto garantiza que cada usuario solo pueda acceder a sus propios datos.

```

    UTILICE LAS OPCIONES DE ARRIBA PARA EDITAR
1  crear política " nombre_política "
2  en " público ". " estudiantes "
3  como PERMISIVO
4  para SELECCIONAR
5  al público
6  usando (
7  id_usuario = auth.uid ( )
8
9  );

```

Ilustración 42 Política de lectura para usuario autenticado en Supabase. Autoría Propia

Paso 42:

En la configuración de la política, establecemos el rol objetivo como autenticado. Esto asegura que solo los usuarios que hayan iniciado sesión en la aplicación puedan ejecutar las acciones permitidas por esta política en la tabla estudiantes.

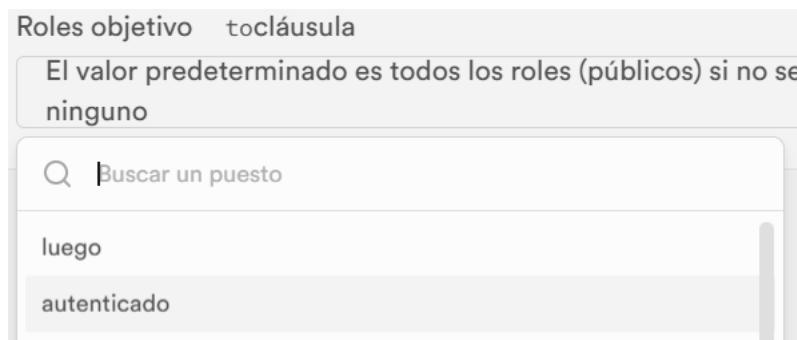


Ilustración 43 Selección de roles objetivo en política de Supabase. Autoría Propia

Paso 43:

Se ha creado correctamente la política para seleccionar en la tabla estudiantes, aplicada únicamente al rol authenticated. Esto garantiza que solo los usuarios autenticados puedan consultar los registros correspondientes.

Ilustración 44 Política creada exitosamente en Supabase. Autoría Propia

Ahora que la política está creada y aplicada correctamente, ya es posible registrar y guardar estudiantes en la base de datos, de manera que al agregarlos desde el formulario, aparecerán en la Lista de estudiantes del panel. Esto confirma que la conexión y los permisos funcionan correctamente.

The screenshot shows the 'Panel de Estudiantes' (Student Panel) interface. At the top, there is a 'Cerrar sesión' (Logout) button. Below it, a 'Registrar estudiante' (Register student) form has three input fields: 'richi richin canayin', 'canayin@gmail.com', and 'computo'. A blue 'Agregar' (Add) button is to the right of the third field. Below the form, a 'Lista de estudiantes' (List of students) section displays two entries: 'richi richin canayin (computo)' and 'David Felipe Roman (computo)'. Underneath this is a 'Subir archivo' (Upload file) section with a dropdown menu labeled 'Selecciona un estudiante:' containing a single entry. It includes a 'Seleccionar archivo' (Select file) button, a message 'Sin archivos seleccionados' (No files selected), and a blue 'Subir' (Upload) button. Finally, a 'Archivos subidos' (Uploaded files) section is shown.

Ilustración 45 Interfaz del Panel de Estudiantes. Autoría Propia

	id identificación UUID	creado... marcas de tie...	nombre texto	correo varchar
<input type="checkbox"/>	6ccfe3e5-7a55-485b-932e-0532e799e951	2025-08-06 20:38:28.260812+C	Richi richin canayin	canayin@gmail.com
<input type="checkbox"/>	81aa1c52-260b-41fb-820b-238be14d8bb3	2025-08-06 20:49:59.626426+C	Pablo	pablo@gmail.com
<input type="checkbox"/>	a2b35ed1-e24e-407f-9372-043236152247	2025-08-06 20:37:36.600917+C	David Felipe Román	felipe@gmail.com

Ilustración 46 usuarios subidos a la base de datos. Autoría Propia

Paso 44:

En el panel lateral de Supabase, haz clic en Storage para iniciar la creación de un bucket donde se guardarán los archivos.

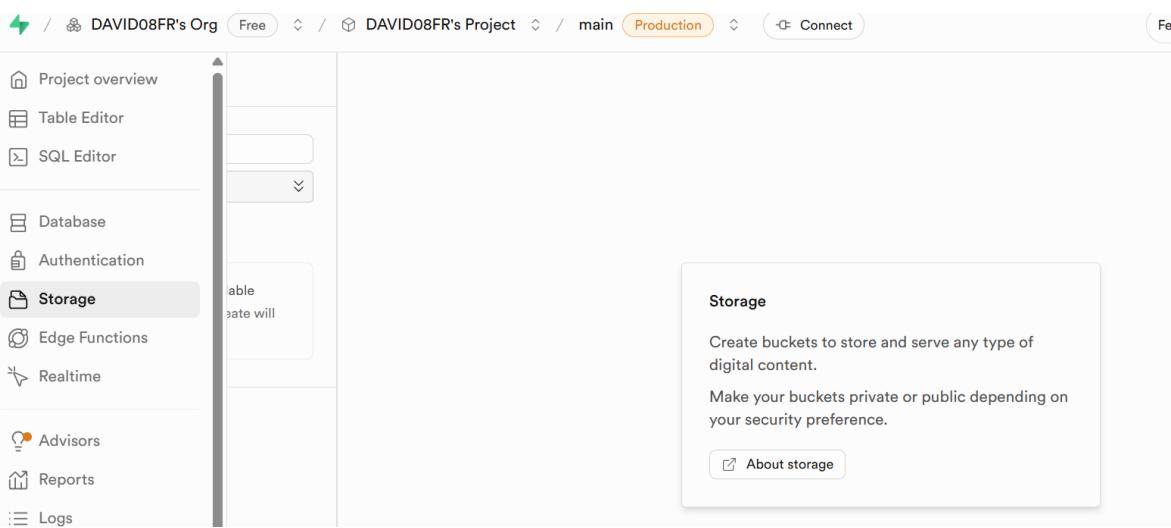


Ilustración 47 Acceder a la sección Storage en Supabase. Autoría Propia

SUBIR ARCHIVOS

Paso 45:

En la ventana Crear un depósito de almacenamiento, escribe el nombre del bucket (por ejemplo, tareas), selecciona Cubo estándar y haz clic en Crear para guardar la configuración.

Crear un depósito de almacenamiento

Nombre del bucket Una vez creados los buckets no se pueden renombrar.
tareas

Sólo letras minúsculas, números, puntos y guiones

Cubo estándar
Compatible con buckets S3.

Cubo de análisis
Almacena archivos Iceberg y está optimizado para cargas de trabajo analíticas.

Esta versión está actualmente en fase alfa y no está disponible para tu proyecto. [Regístrate aquí](#).

Cubo público
Cualquiera puede leer cualquier objeto sin ninguna autorización.

Restricciones adicionales

[Cancelar](#) [Crear](#)

Ilustración 48 Crear un bucket de almacenamiento. Autoría Propia

Una vez creado el bucket llamado tareas, accede a él para comenzar a subir archivos o crear carpetas según sea necesario.

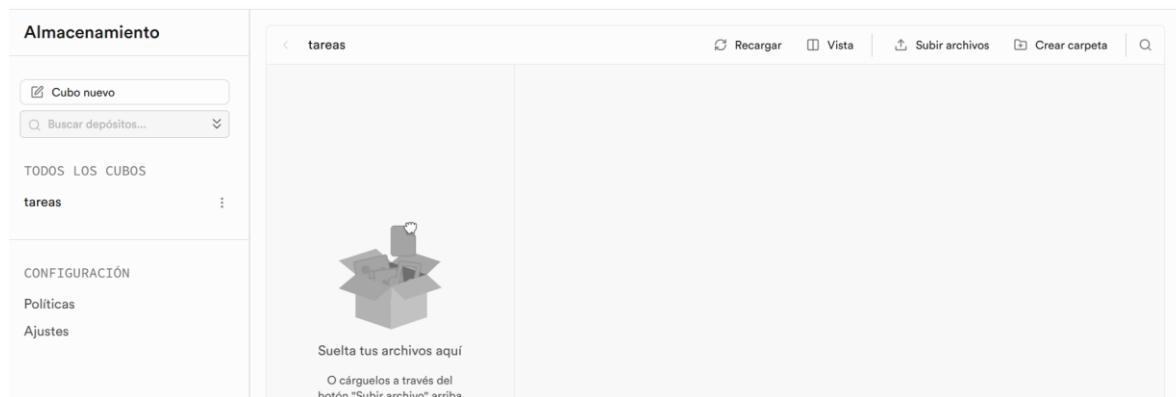


Ilustración 49 Acceder al bucket creado. Autoría Propia

Paso 47:

En la sección Políticas de almacenamiento, selecciona el bucket tareas y haz clic en Nueva política para definir las reglas de acceso y permisos para los usuarios según las necesidades del proyecto.

The screenshot shows the 'Almacenamiento' (Storage) section of the Google Cloud Platform. On the left, there's a sidebar with 'Cubo nuevo' and a search bar for 'Buscar depósitos...'. Below that are sections for 'TODOS LOS CUBOS' and 'tareas'. Under 'CONFIGURACIÓN', the 'Políticas' tab is selected, showing 'Ajustes'. The main content area is titled 'Políticas de almacenamiento' and contains three sections: 'tareas' (with a 'Nueva política' button), 'Otras políticas en storage.objects' (with a 'Nueva política' button), and 'Políticas en storage.buckets' (with a 'Nueva política' button). A note at the top says: 'Proteja sus archivos con políticas que definen las operaciones permitidas para sus usuarios a nivel de depósito.' Below each section, it says 'Aún no se han creado políticas'.

Ilustración 50 Configurar políticas de almacenamiento. Autoría Propia

Paso 48:

En la ventana para agregar una nueva política al bucket tareas, selecciona si deseas empezar rápidamente usando una plantilla predefinida o crear una política desde cero para personalizar completamente las reglas de acceso y permisos.



Ilustración 51 Agregar nueva política al bucket. Autoría Propia

Paso 49:

En esta sección, asigna un nombre descriptivo a la política (por ejemplo, Permitir acceso). Luego, selecciona las operaciones permitidas, como seleccionar, descargar y listar, según lo que necesites habilitar para los usuarios o roles que tendrán acceso al bucket.

Nombre de la póliza
Un nombre descriptivo para su póliza

Permitir acceso 0 / 50

Operación permitida
Según las operaciones que haya seleccionado, puede utilizar las funciones resaltadas en la [biblioteca del cliente](#).
 SELECCIONAR INSERTAR ACTUALIZAR BORRAR
 subir descargar lista actualizar mover Copiar eliminar
 crear URL firmada crear URLs firmadas obtenerPublicUrl

Roles objetivo
Aplicar la política a los roles seleccionados

Ilustración 52 Configurar permisos de la política. Autoría Propia

Paso 50:

Selecciona el rol authenticated para que solo los usuarios autenticados puedan acceder. En la sección Policy definition, agrega la condición SQL que limita el acceso al bucket tareas únicamente al propietario del recurso:

authenticated x

Policy definition 1 bucket_id = 'tareas' and auth.uid() = owner
Provide a SQL conditional expression that returns a boolean.

Ilustración 53 Definir roles y condición de acceso. Autoría Propia

Paso 51:

En el panel Subir archivo, seleccionamos al estudiante, elegimos el archivo desde nuestro equipo y damos clic en Subir. Confirmamos que el archivo aparece en Archivos subidos con sus enlaces para ver/descargar, según las políticas configuradas.

The screenshot displays four main sections of a mobile application:

- Registrar estudiante**: A form with fields for Nombre del estudiante, Correo, and Clase, and a blue "Agregar" button.
- Lista de estudiantes**: A list of three students: pablo (computo), richi richin canayin (computo), and David Felipe Roman (computo).
- Subir archivo**: A section to select a student and upload a file named 2025-05-11 20-16-15.mkv, with a green "Subir" button.
- Archivos subidos**: A list of uploaded files including Asistencias Back y front.rar, DAVID FELIPE ROMAN.pdf, fbd9237ee2fa47200e6c39f1fb1a2b40.jpg, and 2025-05-11 20-16-15.mkv, with a thumbnail of a silver car.

Ilustración 54 Subir archivos desde la aplicación. Autoría Propia

Paso 52:

En el panel de Almacenamiento de Supabase, dentro del cubo tareas, se confirma que todos los archivos subidos desde la app están correctamente almacenados. Esto asegura que la integración de carga funciona y que las políticas permiten el acceso según lo configurado.

The screenshot shows the Supabase storage interface with the following details:

- Almacenamiento** sidebar with options: Cubo nuevo, Buscar depósitos..., TODOS LOS CUBOS, tareas, and CONFIGURACIÓN.
- Cubo** view for bucket **a8d8c271-3cce-4c57-ba72-29483ec6f395**:

		Recargar
<input checked="" type="checkbox"/>	a8d8c271-3cce-4c57-ba72-29483ec6f395	[refresh icon]
<input checked="" type="checkbox"/>	2025-05-11 20-16-15.mkv	
<input checked="" type="checkbox"/>	Asistencias atrás y adelant...	
<input checked="" type="checkbox"/>	CONSTANCIA.pdf	
<input checked="" type="checkbox"/>	DAVID FELIPE ROMÁN.pdf	
<input checked="" type="checkbox"/>	fbd9237ee2fa47200e6c39...	
<input checked="" type="checkbox"/>	trivias.docx	

Ilustración 55 Verificación de archivos en el almacenamiento. Autoría Propia

ACTUALIZAR Y ELIMINAR REGISTROS.

En esta sección del código se agregan dinámicamente los botones Editar y Eliminar para cada registro de estudiante cargado en la lista.

- El botón Editar (btnEditar) permite abrir los datos del estudiante en el formulario para su modificación, utilizando la función editarEstudiante(est).
- El botón Eliminar (btnEliminar) ejecuta la función eliminarEstudiante(est.id) para borrar el registro seleccionado.

Ambos botones se generan con clases de Bootstrap para darles estilo (btn-outline-primary para editar y btn-outline-danger para eliminar) y se añaden a la interfaz junto con la información del estudiante.

```
43  async function cargarEstudiantes() {
61    data.forEach((est) => {
71
72      const btnEditar = document.createElement("button");
73      btnEditar.type = "button";
74      btnEditar.className = "btn btn-sm btn-outline-primary me-2";
75      btnEditar.innerHTML = `<i class="fas fa-edit me-1"></i>Editar`;
76      btnEditar.addEventListener("click", () => editarEstudiante(est));
77
78      const btnEliminar = document.createElement("button");
79      btnEliminar.type = "button";
80      btnEliminar.className = "btn btn-sm btn-outline-danger";
81      btnEliminar.innerHTML = `<i class="fas fa-trash-alt me-1"></i>Eliminar`;
82      btnEliminar.addEventListener("click", () => eliminarEstudiante(est.id));
83
84      acciones.appendChild(btnEditar);
85      acciones.appendChild(btnEliminar);
86
87      li.appendChild(info);
88      li.appendChild(acciones);
89      lista.appendChild(li);
90    });
91  }
```

Ilustración 56 Implementación de botones "Editar" y "Eliminar" en dashboard.js

Este fragmento de código HTML define la sección de **Registro / Edición** de estudiantes. Incluye:

- **Campos de entrada** para capturar el **nombre**, **correo** y **clase** del estudiante.
- Un **botón principal** (btn-guardar) que ejecuta la función agregarEstudiante() cuando se hace clic, permitiendo añadir el registro a la base de datos.
- Estructura y estilos proporcionados por **Bootstrap**, con clases como form-control para los campos y btn btn-primary para el botón.
- Iconos de **Font Awesome** para mejorar la interfaz visual y hacer más intuitivas las acciones.



```
2 <html lang="es">
25 <body>
40   <main class="container py-4 py-lg-5">
41     <div class="row g-4">
43       <div class="col-lg-4">
44         <div class="sticky col-12">
45           <!-- Registrar / Editar -->
46           <section class="panel mb-4">
47             <div class="panel-header">
48               <i class="fas fa-user-plus text-primary"></i>
49               <span class="section-title">Registrar estudiante</span>
50             </div>
51             <div class="panel-body">
52               <div class="mb-3">
53                 <label for="nombre" class="form-label">Nombre</label>
54                 <input type="text" id="nombre" class="form-control" placeholder="Nombre del estu">
55               </div>
56               <div class="mb-3">
57                 <label for="correo" class="form-label">Correo</label>
58                 <input type="email" id="correo" class="form-control" placeholder="correo@ejemplo">
59               </div>
60               <div class="mb-3">
61                 <label for="clase" class="form-label">Clase</label>
62                 <input type="text" id="clase" class="form-control" placeholder="Ej. Matemáticas">
63               </div>
64               <button id="btn-guardar" onclick="agregarEstudiante()" class="btn btn-primary w-100">
65                 <i class="fas fa-plus-circle me-2"></i>Agregar
66               </button>
67             </div>
68           </section>
69         </div>
70       </div>
71     </div>
72   </main>
73 </body>
74 </html>
```

Ilustración 57 Formulario de registro y edición de estudiantes en el archivo dashboard.html

Posterior a ello debemos entrar a supabase y crear las políticas necesarias en Supabase para permitir que cada usuario únicamente pueda actualizar y eliminar los registros que le pertenecen en la tabla estudiantes. Esto se logra activando la Seguridad a Nivel de Fila (RLS) y definiendo condiciones donde el campo user_id coincida con el identificador del usuario autenticado (auth.uid()), garantizando así la integridad y seguridad de la información.



Ilustración 58 Configuración de políticas RLS para actualizar y eliminar registros

En esta imagen se muestra la configuración de una política de seguridad en Supabase para permitir la actualización de registros en la tabla estudiantes.

La política está configurada bajo el comando ACTUALIZAR y condicionada para que solo el usuario propietario del registro pueda modificarlo, verificando que el campo user_id coincida con el identificador del usuario autenticado (auth.uid()). Esto asegura que cada usuario únicamente pueda editar sus propios datos.

x Política de actualización: Política para actualizar

Ver detalles de la póliza ▾

Política para actualizar público . estudiantes

Comportamiento político ascláusula

Permisivo

Comando de política forcláusula

SELECCIONA INSERTAR ACTUALIZAR BORRAR TODO

Roles objetivo tocláusula

El valor predeterminado es todos los roles (públicos) si no se selecciona ninguno

🔒 UTILICE LAS OPCIONES DE ARRIBA PARA EDITAR

```
1 alterar política " Política para actualizar "
2 en " público ". " estudiantes "
5 al público
6 usando (
7   ( user_id = auth . uid ( ) )
8 con cheque (
9   (user_id = auth.uid())
10 );
```

Ilustración 59 Creación de política RLS para actualizar registros

En esta imagen se observa la configuración de una política en Supabase para la acción BORRAR en la tabla estudiantes.

La política establece que únicamente se puedan eliminar los registros cuyo campo user_id coincide con el identificador del usuario autenticado (auth.uid()).

Esto garantiza que cada usuario solo pueda borrar sus propios registros, evitando la eliminación de información perteneciente a otros usuarios.

The screenshot shows the Supabase RLS (Row-Level Security) policy creation interface. The policy is named "Eliminar". It is defined for the "público . estudiantes" role. The "Comportamiento político" is set to "Permisivo". Under "Comando de política", the "Borrar" option is selected. In the "Roles objetivo" section, it says "El valor predeterminado es todos los roles (públicos) si no se selecciona ninguno". Below the interface, there is a code editor with the following SQL-like pseudocode:

```
1 UTILICE LAS OPCIONES DE ARRIBA PARA EDITAR
2 modificar la política " Eliminar "
3 en " público ". " estudiantes "
4 al público
5 usando (
6   user_id = auth . uid ( )
7 );
8 ;
```

Ilustración 60 Creación de política RLS para eliminar registros

Una vez creadas las políticas en Supabase y modificados los archivos .js y .html para implementar las funciones de actualización y eliminación, es necesario ingresar a GitHub Desktop para guardar los cambios en el repositorio.

En esta etapa se seleccionan los archivos modificados, se añade un mensaje descriptivo en el campo Summary y se realiza el Commit. Posteriormente, se debe ejecutar la opción Push para enviar los cambios al repositorio remoto, asegurando que el proyecto actualizado esté disponible en GitHub.

```

Changes 3 History
Filter
3 changed files
✓ dashboard.html
✓ js\dashboard.js
✓ styles.css

Summary (required)
Description
Commit 3 files to main

```

Changes 3 History

Filter

3 changed files

✓ dashboard.html

✓ js\dashboard.js

✓ styles.css

Summary (required)

Description

Commit 3 files to main

```

@@ -1,298 +1,116 @@
- // =====
- // Configuración Supabase
- // =====
- const SUPABASE_URL = "https://fzbheuoaxnenhrwadrvn.supabase.co";
- const SUPABASE_KEY = "eyJhbGciOiJIUzI1NiIsInR5cCI6Ikpc3Mi0iJzdXBhYmFzZSIsInJlZiI6Im
Z6YmhldwSheG51bmhyd2FkcnZuiwicm9sZSI6InFub24iLCJpYXQiOjE3NTQ1MDUwOTMsImV4cCI6MjA3MDAA4MTASM3
0.49jf_IMBvKt-HWpmN-9USgw6tDqUKzrWluoQnFi40"; // clave de ejemplo
-
- // Crear cliente de Supabase
- const client = supabase.createClient(SUPABASE_URL, SUPABASE_KEY);
-
- // =====
- // Estado de edición
- // =====
- let editId = null;
-
- // =====
- /** Helpers */
- // =====
- function limpiarFormulario() {
-   document.getElementById("nombre").value = "";
-   document.getElementById("correo").value = "";
-   document.getElementById("clase").value = "";
- }
-
- // =====
- /** Crear / Actualizar estudiante */
- // =====
- async function agregarEstudiante() {

```

Ilustración 61 Commit y Push de los cambios en GitHub Desktop

Una vez realizado el Commit y Push en GitHub, debemos acceder a la página de Vercel, donde está desplegado nuestro proyecto.

Vercel detectará automáticamente los cambios en el repositorio y comenzará el proceso de actualización.

Debemos esperar a que la compilación finalice para que el enlace del proyecto se actualice con la nueva versión, reflejando las modificaciones realizadas en el código.

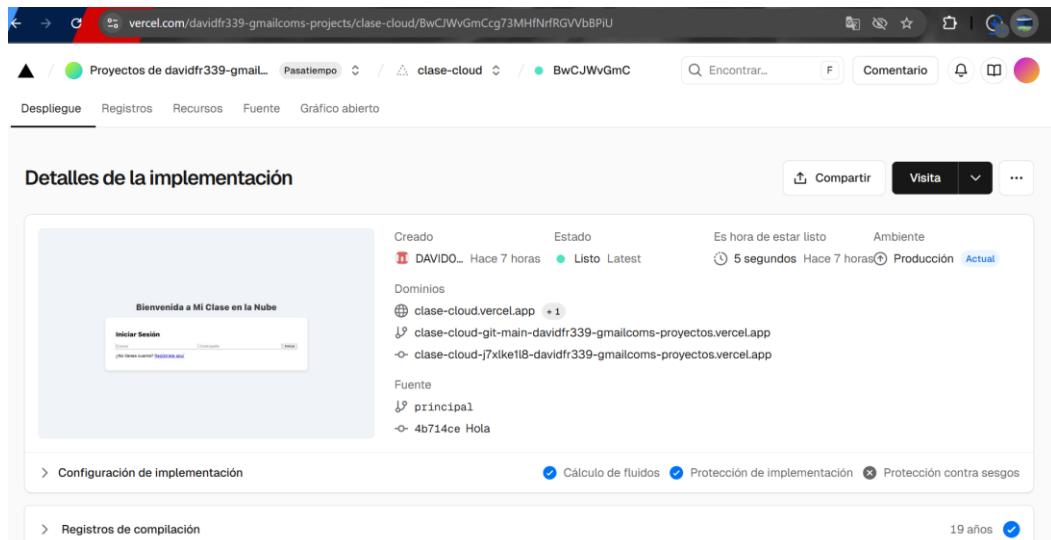


Ilustración 62 Actualización automática del proyecto en Vercel

Ingresamos al enlace del proyecto desplegado en Vercel y procedemos a iniciar sesión en la aplicación.

Para ello, introducimos el correo electrónico y contraseña registrados previamente, y presionamos el botón "Entrar".

Este paso nos permite acceder a las funcionalidades del sistema y verificar que las actualizaciones realizadas se han aplicado correctamente.

Bienvenida a Mi Clase en la Nube

The image shows a login form titled "Iniciar Sesión". It has two input fields: one for "Nombre" (david08@gmail.com) and one for "Contraseña" (.....). A "Entrar" button is to the right. Below the fields is a link "¿No tienes cuenta? [Regístrate aquí](#)".

Ilustración 63 Inicio de sesión en la aplicación web

Una vez actualizado el proyecto en Vercel, ingresamos a la liga generada y procedemos a iniciar sesión con nuestras credenciales registradas. Al acceder, se muestra el Panel de Estudiantes, donde es posible visualizar la lista de estudiantes registrados junto con las opciones para editar o eliminar cada registro,

The image shows the "Panel de Estudiantes" interface. On the left, there is a "Registrar estudiante" form with fields for Nombre (Nombre del estudiante), Correo (correo@ejemplo.com), and Clase (Ej. Matemáticas), and a blue "Agregar" button. On the right, there is a "Lista de estudiantes" section displaying three student entries: "ivan second (mate) — ivan@gmail.com", "ivan second (computo) — ivan@gmail.com", and "david (historia) — davidfr45@gmail.com", each with "Editar" and "Eliminar" buttons. Below this is an "Archivos subidos" section showing "Sin archivos". At the top right, there are "Sesión activa" and "Cerrar sesión" buttons.

Ilustración 64 Acceso a la plataforma y panel de estudiantes

Al hacer clic en el botón **Editar** junto a un estudiante, los datos actuales se cargan en el formulario de registro. Esto nos permite modificar campos como **nombre**, **correo** y **clase**. Una vez realizados los cambios, presionamos **Actualizar** para guardar la información en la base de datos, reflejando los cambios en la lista de estudiantes en tiempo real.

Ilustración 65 Actualización de datos de un estudiante

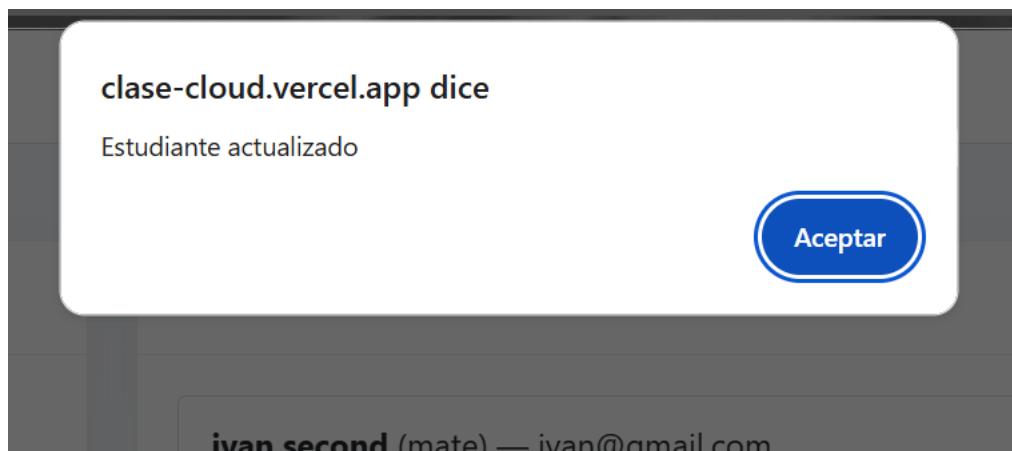


Ilustración 66 Mensaje estudiante actualizado

Como se aprecia también en la base de datos se actualizan los dato modificados desde el sitio.

<input type="checkbox"/>	81aa1c52-260b-41fb-820b-238be14d8bb3	pablo@gmail.com	a8d8c271-3cce-4c57-ba72-29483ec6f395	computo
<input type="checkbox"/>	9bc000ac-69bb-4d61-a2a4-b90d7852266	man@gmail.com	303bcef3-ca15-48f2-8fbc-1eac7c1d956b	ciencias
<input type="checkbox"/>	a2b35ed1-e24e-407f-9372-043236152247	felipe@gmail.com	a8d8c271-3cce-4c57-ba72-29483ec6f395	computo

Ilustración 67 Datos actualizados en las base de datos

Al seleccionar el botón Eliminar en la lista de estudiantes, el sistema despliega un cuadro de diálogo solicitando confirmación para borrar el registro. Este mensaje asegura que el usuario valide su intención antes de realizar la eliminación definitiva, evitando borrados accidentales. Si se presiona Aceptar, el estudiante será eliminado de la base de datos; si se presiona Cancelar, la operación será abortada.

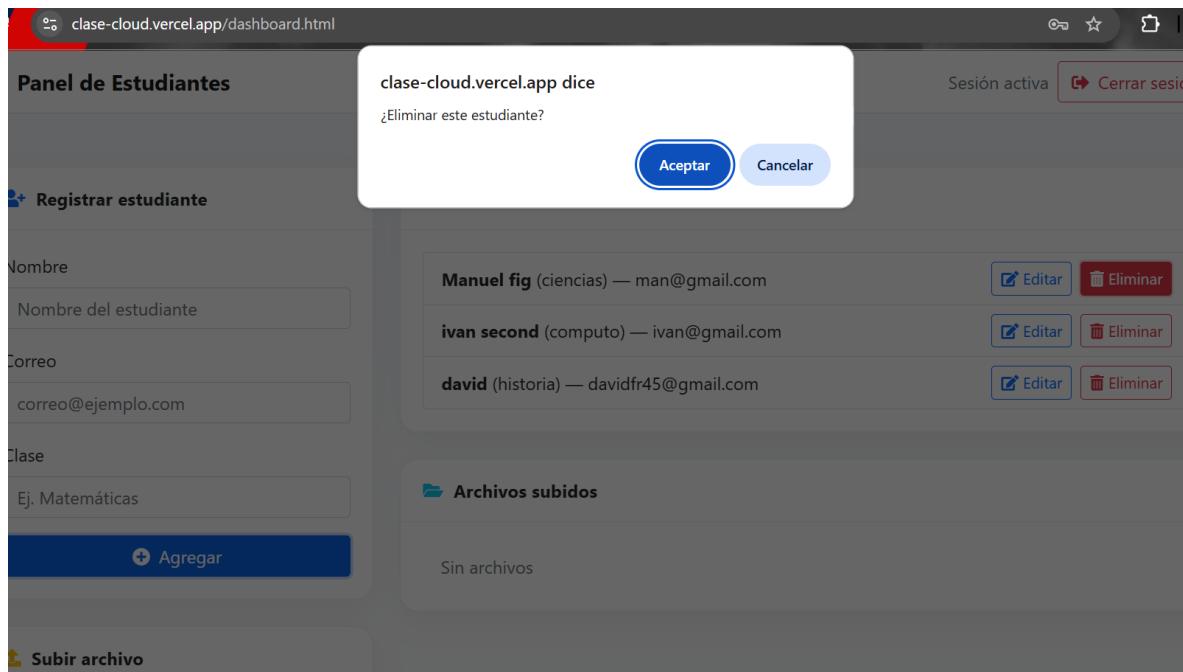


Ilustración 68 Confirmación de eliminación de estudiante

En esta etapa se muestra el resultado posterior a la eliminación de un usuario en el Panel de Estudiantes. Tras confirmar la acción en el cuadro de diálogo, el sistema actualiza automáticamente la lista, retirando el registro correspondiente y manteniendo visibles únicamente los estudiantes activos.

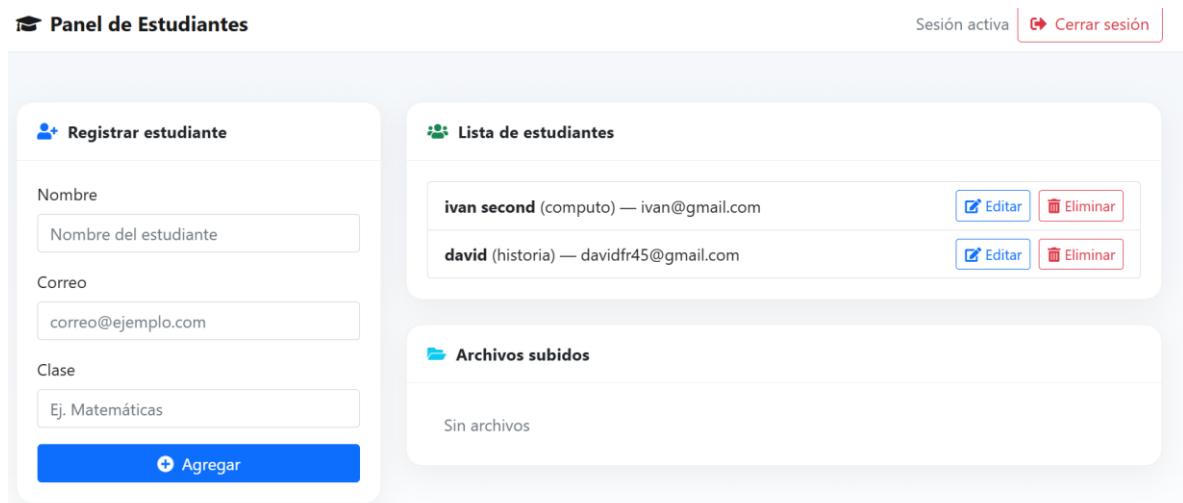


Ilustración 69 Eliminación de usuario en el Panel de Estudiantes

Y como se aprecia en la imagen también se borran los datos del estudiante desde la base de datos .

	<code>id</code> <small>uuid</small>	<code>correo</code> <small>varchar</small>	<code>user_id</code> <small>uuid</small>	<code>clase</code> <small>text</small>
	0126108d-9dec-415f-8b80-1feb0e6072d0	ivan@gmail.com	303bcef3-ca15-48f2-8fbc-1eac7c1d956b	computo
	106a93ce-e645-4808-a6e9-86dfd771f0dC	davi@gmail.com	5da645b6-fb7a-4e53-ac30-5abae96656d	historia
	6ccfe3e5-7a55-485b-932e-0532e799e95l	canayin@gmail.com	a8d8c271-3cce-4c57-ba72-29483ec6f395	computo
	81aa1c52-260b-41fb-820b-238be14d8bb3	pablo@gmail.com	a8d8c271-3cce-4c57-ba72-29483ec6f395	computo
	a2b35ed1-e24e-407f-9372-043236152247	felipe@gmail.com	a8d8c271-3cce-4c57-ba72-29483ec6f395	computo
	cf6377ab-7b65-4d37-8005-1db5c8846b3c	davidfr45@gmail.com	303bcef3-ca15-48f2-8fbc-1eac7c1d956b	historia

Ilustración 70 Datos eliminado en la base de datos

Análisis de la práctica

La práctica permitió experimentar un flujo de trabajo real en el desarrollo de aplicaciones web modernas utilizando servicios en la nube.

En la primera fase, se gestionó el código fuente con GitHub, lo que facilitó el control de versiones y la colaboración. Luego, mediante Vercel, se automatizó el despliegue continuo, asegurando que cada cambio en el repositorio se reflejara en la versión en línea de la aplicación. Esto redujo tiempos de publicación y eliminó procesos manuales de despliegue.

En la segunda fase, con Supabase, se configuró un sistema completo de backend sin necesidad de infraestructura propia. Se implementó la autenticación segura de usuarios, la gestión de bases de datos relacionales y el almacenamiento de archivos, todo controlado mediante políticas de acceso para garantizar la seguridad de la información.

En conjunto, la práctica integró de manera eficiente herramientas SaaS y BaaS para cubrir las necesidades de desarrollo frontend, backend y despliegue, reforzando conocimientos clave sobre computación en la nube, trabajo colaborativo y seguridad en entornos de producción.