

**ESCUELA SUPERIOR POLITECNICA DEL LITORAL**

**EXAMEN SEGUNDO PARCIAL LABORATORIO ADMINISTRACIÓN DE SISTEMAS Y SERVICIOS DE RED**

**GRUPO 5**

**INTEGRANTES**

TARANTO USEY JOSHUA ISRAEL

VARGAS ORELLANA LOURDES JESSENIA

**TUTORA**

MSIG. ADRIANA COLLAGUAZO JARAMILLO

**PARALELO 101**

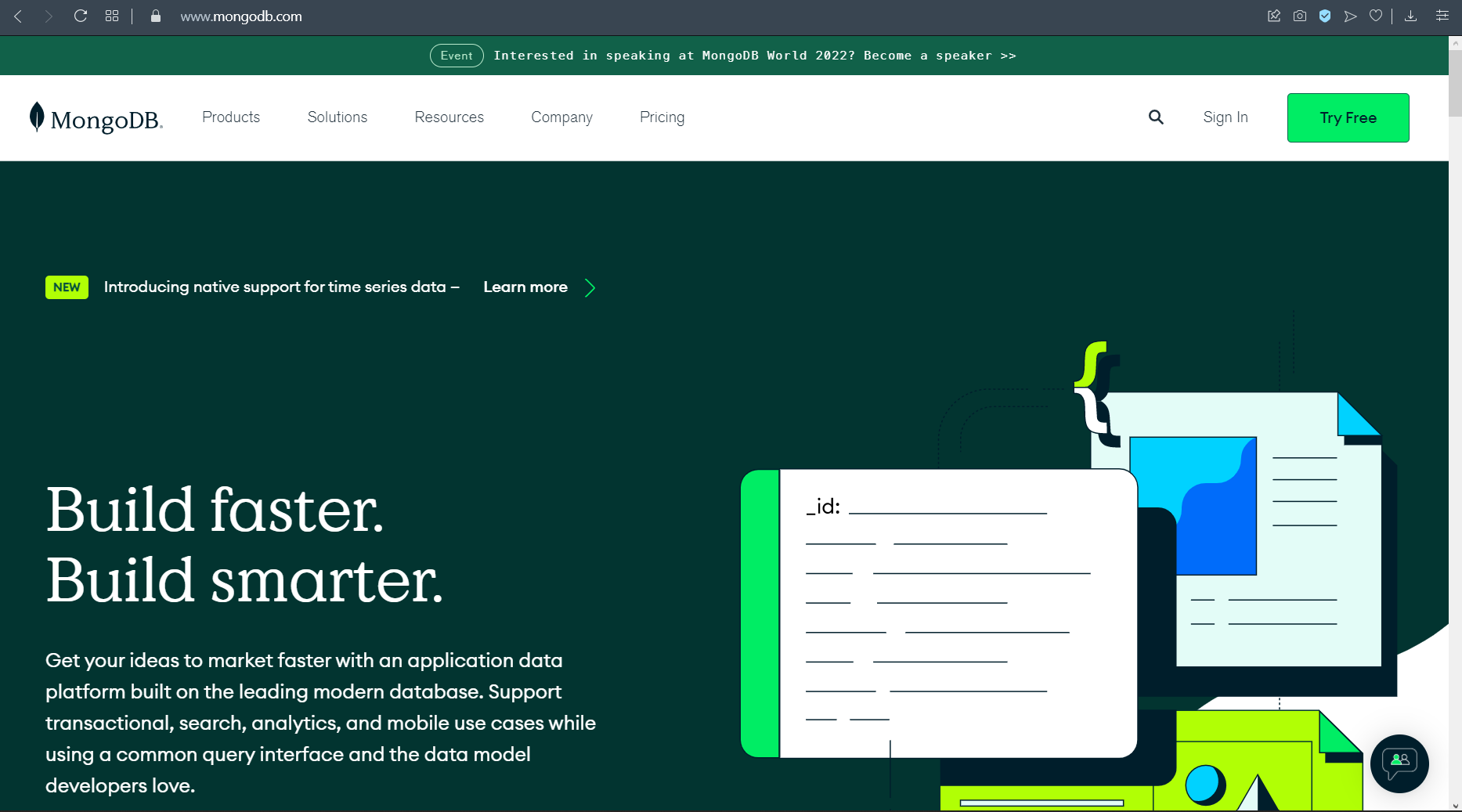
**PAO II 2021**

**ESCENARIO**

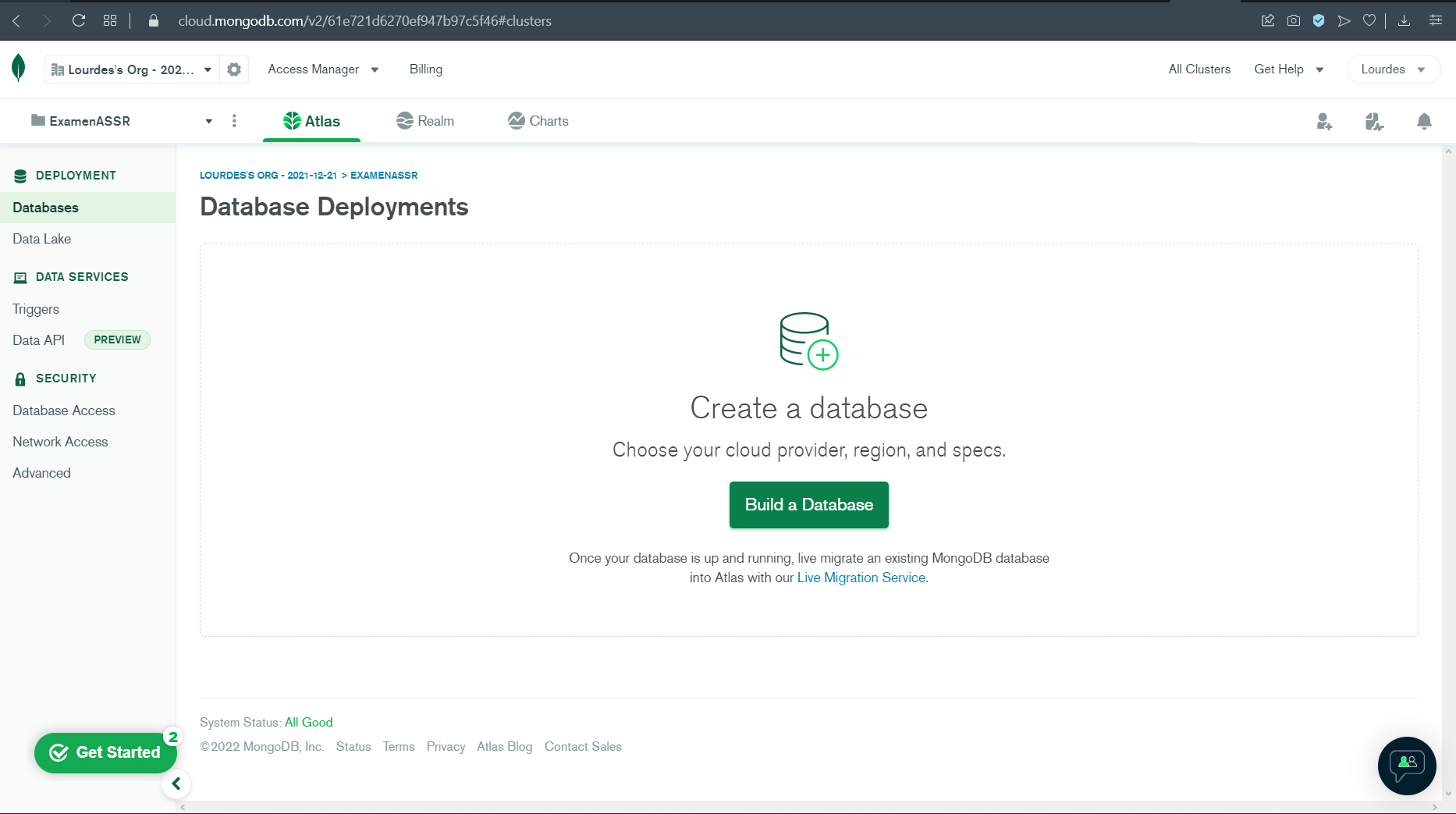
La empresa AditaCar S.A. lo ha contratado a Usted como desarrollador web. La nueva plataforma web será desarrollada en el backend usando Node.js y en el frontend con REACT. El contenido de la plataforma web mostrará la ubicación de las estaciones de carga para vehículos eléctricos. El principal objetivo de la página web es lograr el incremento del 25% al 40% en el segmento del mercado de clientes que han comprado vehículos eléctricos.

**TEMA 1: BASE DE DATOS (25 PUNTOS)**

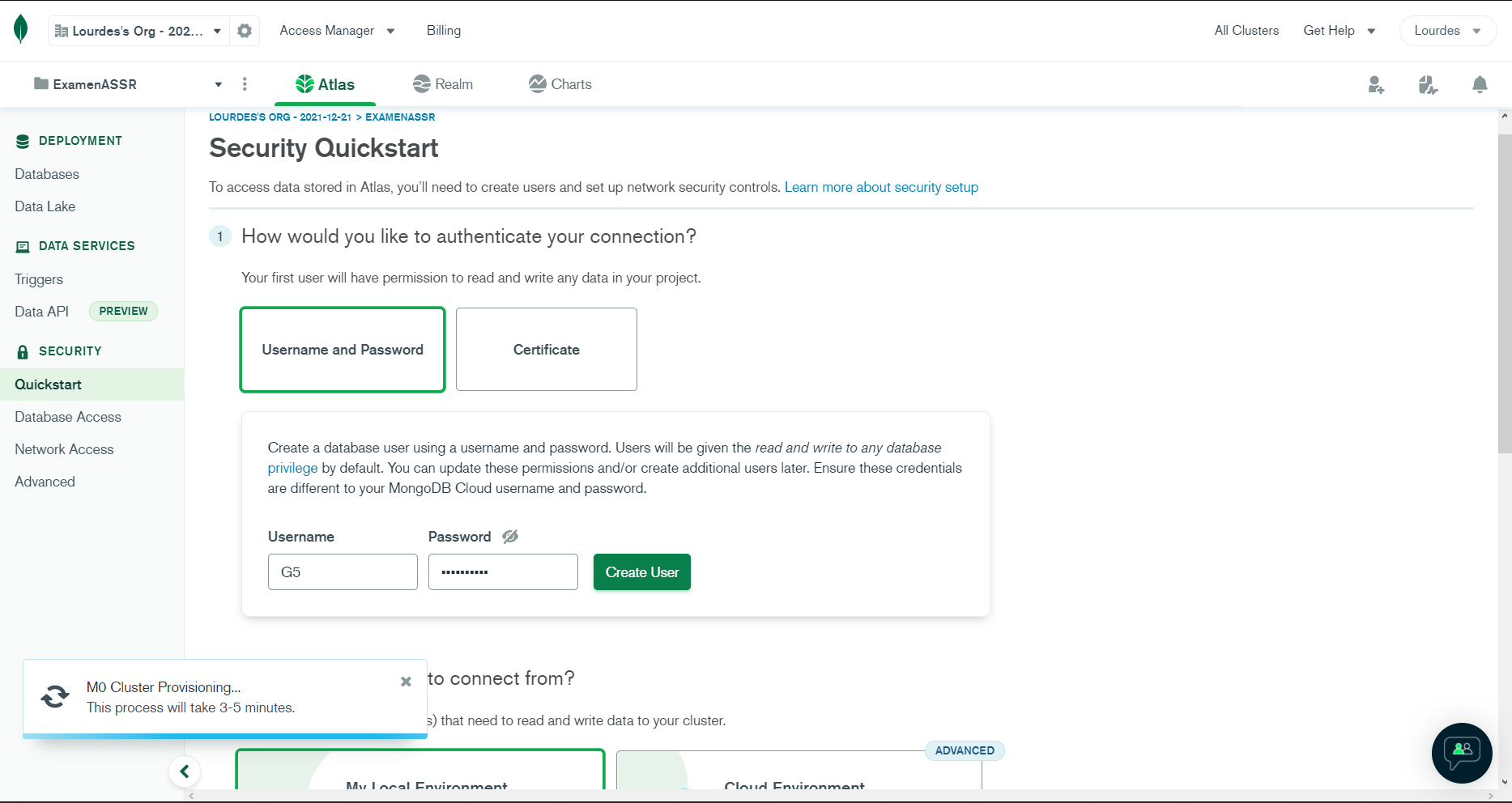
1. Creamos una base de datos no relacional accediendo a <https://www.mongodb.com>



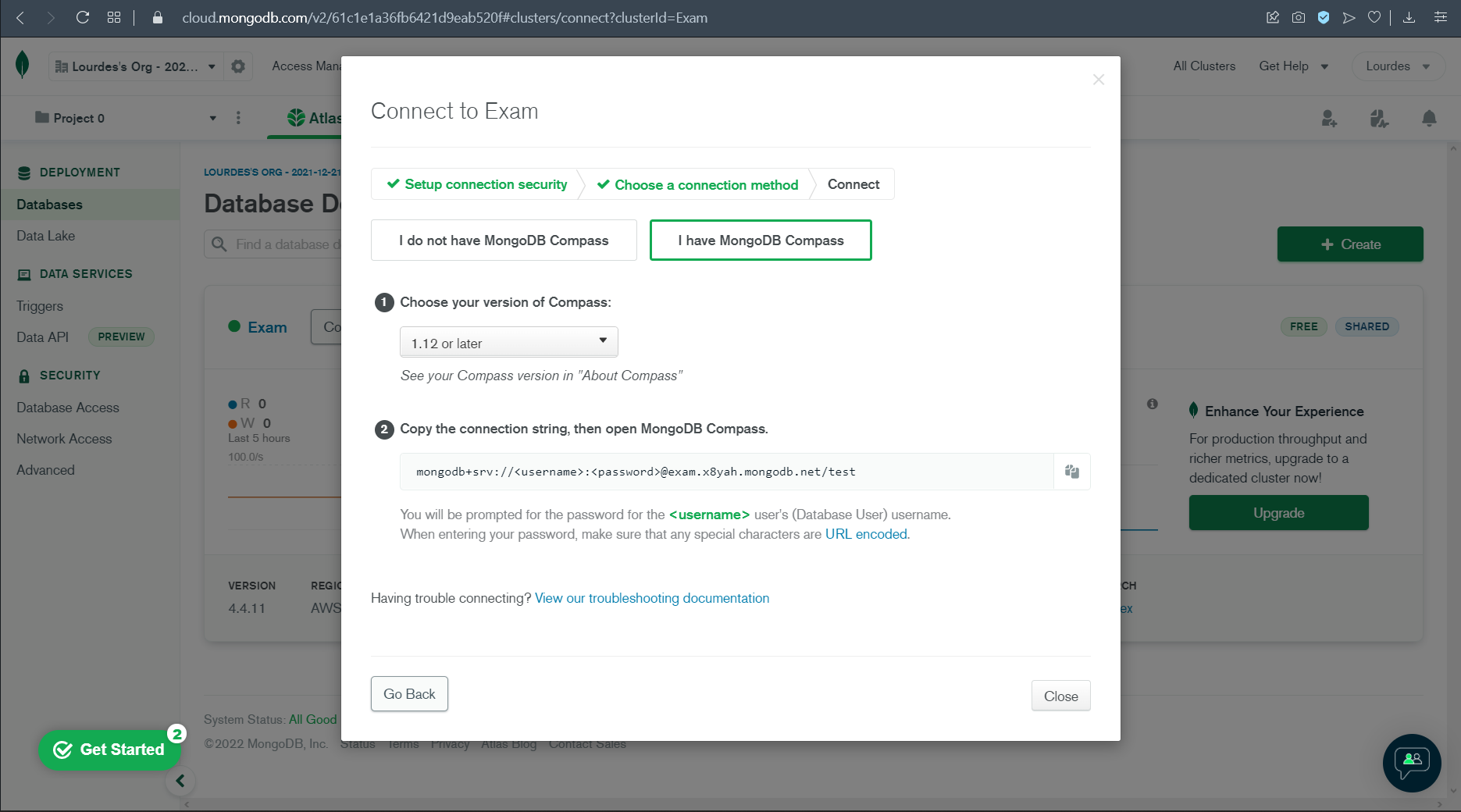
1. Creamos una cuenta gratuita con nuestro usuario de espol
2. Una vez creado nuestro usuario creamos una Database nueva

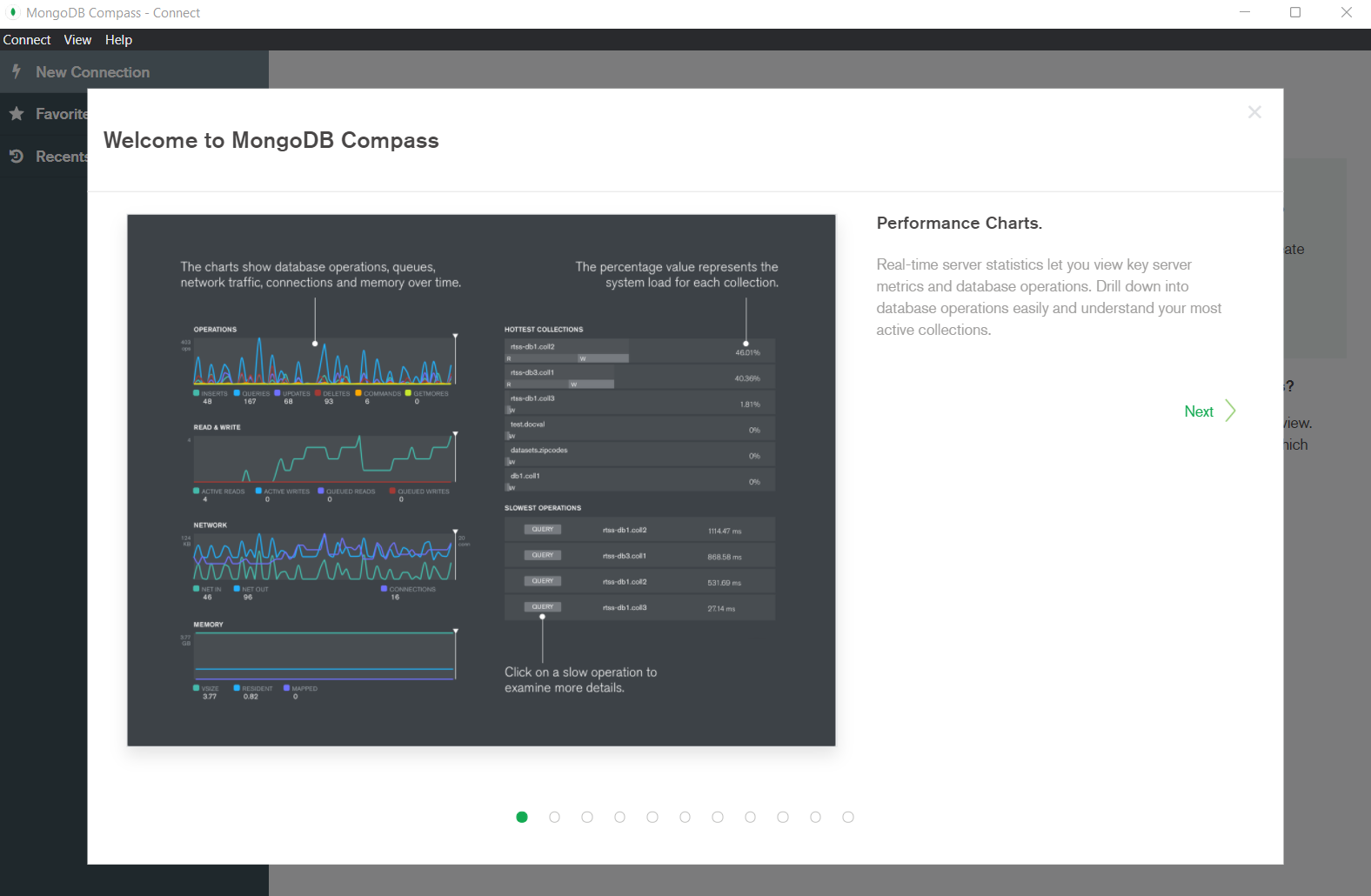


1. Seguimos las instrucciones que nos ofrece para hacerlo de forma eficiente.
2. Creamos un usuario

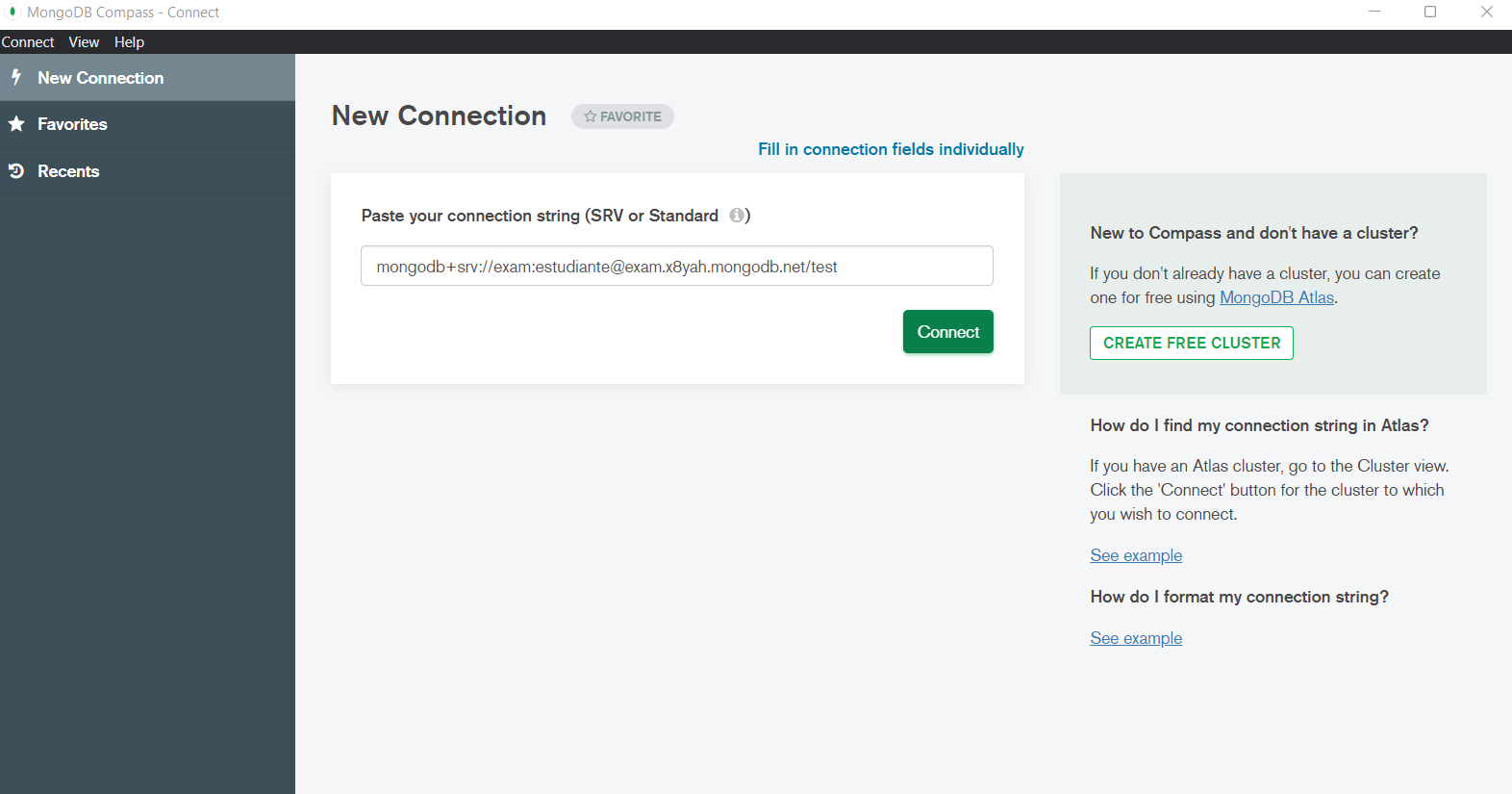


1. Una vez creada la base de datos vamos a conectarnos, seguimos las instrucciones indicadas.
2. Si preferimos podemos descargar MongoDB Compass que nos ayudará a conectarnos con nuestra base de datos

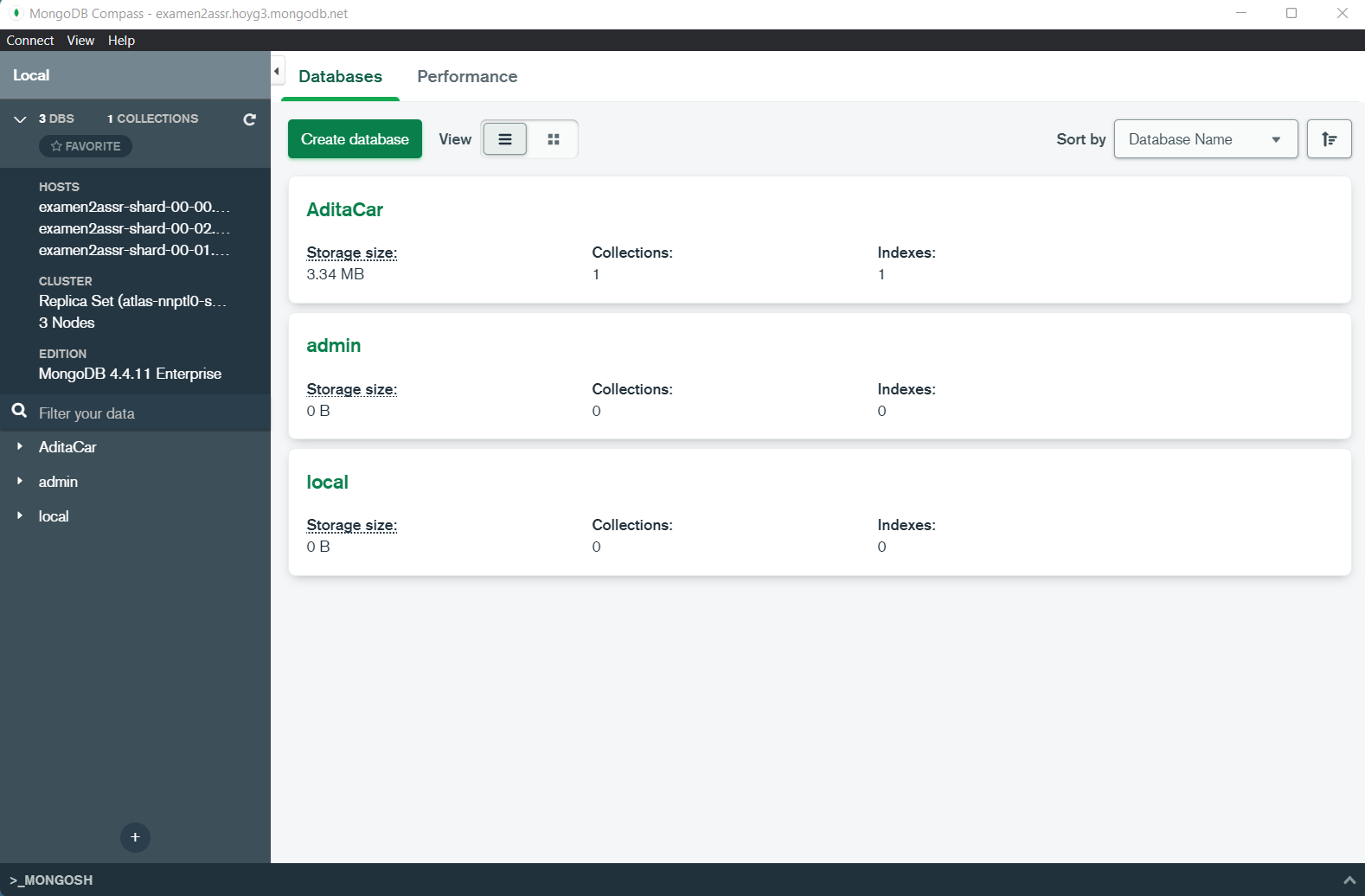




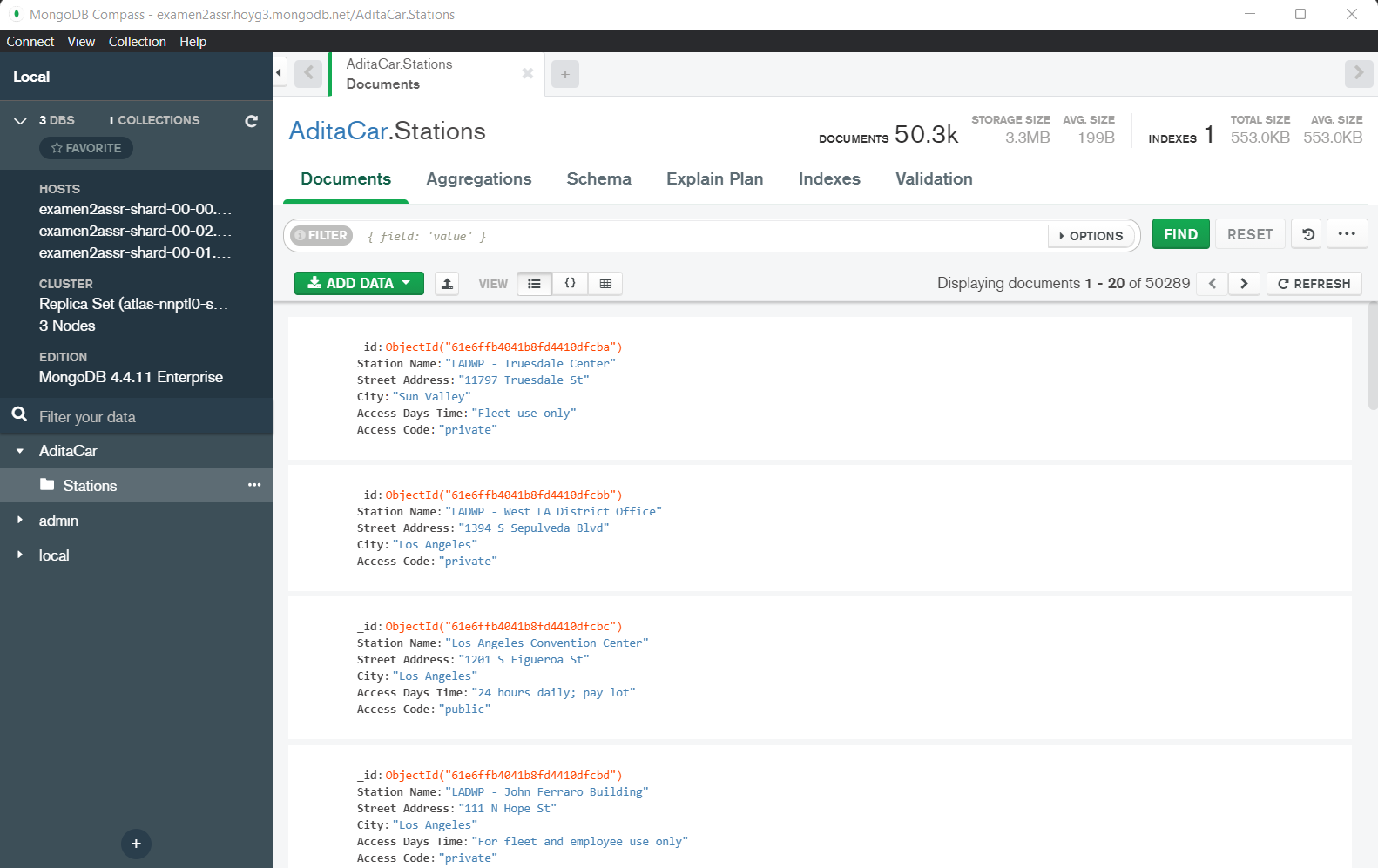
1. Establecemos la coneccion, sustituimos en los campos userid y password nuestro usuario y contraseña creado para acceder a la BD.



1. Si se conecta correctamente podemos observar nuestra base de datos, si se importaron correctamente el archivo .csv podremos apreciar su contenido.



1. Nos desplazamos hasta el nombre AditaCar.Stations y podemos visualizar los datos contenidos.

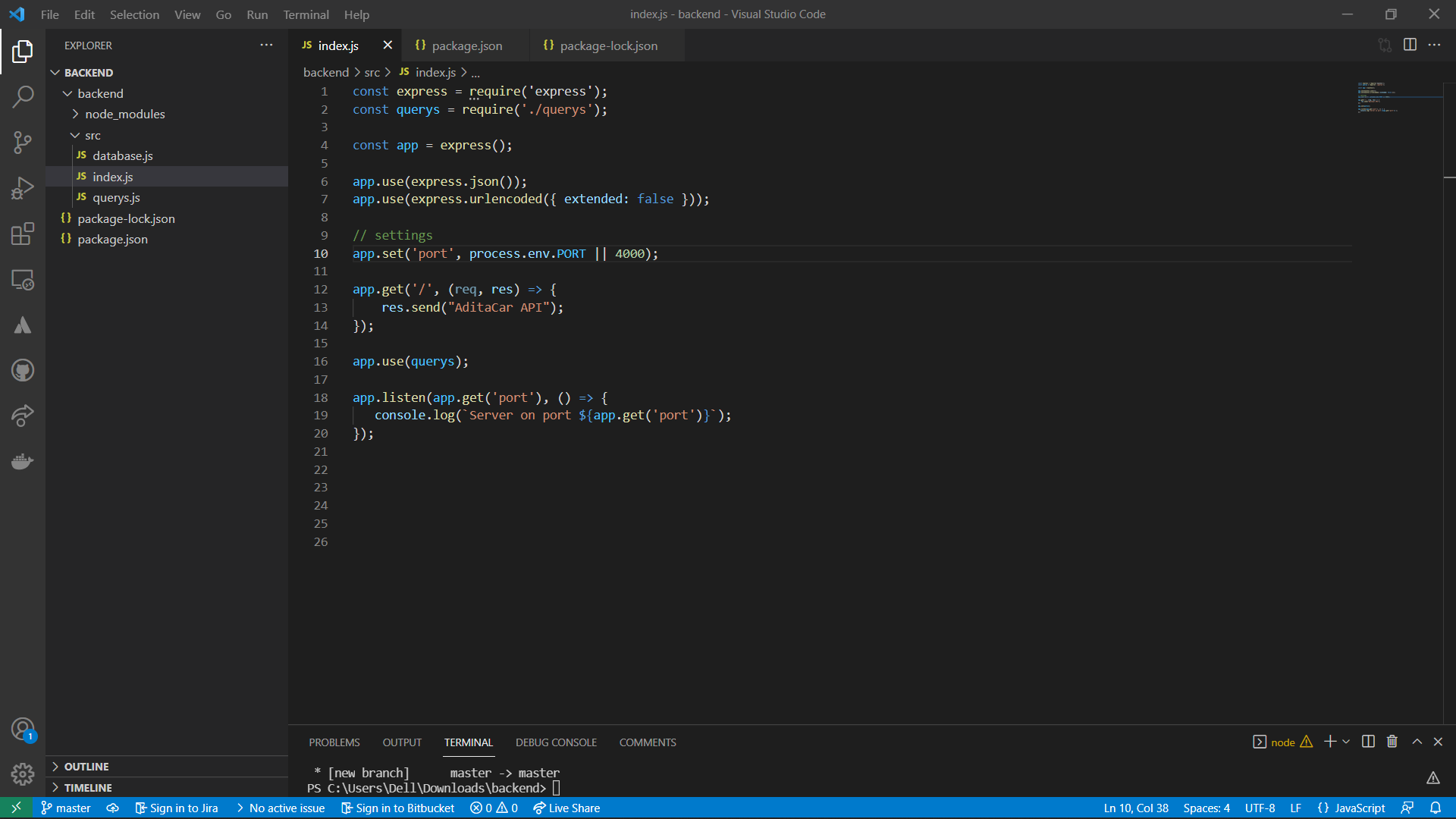


**TEMA 2: DESARROLLO WEB (45 PUNTOS)**

1. creamos un nuevo proyecto desde el cual construiremos nuestro **backend** (link de acceso al GitHub: <https://github.com/joistaus/ASSR-G5> )
2. Nos conectamos a la base de datos



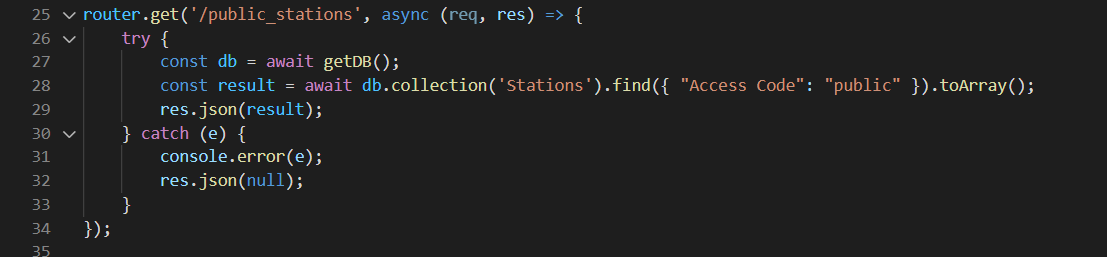
1. Es el archivo index.js especificamos el como acceder y conectarnos al api para que esta luego pueda ser desplegada de acuerdo a nuestra elección en este caso mediante Heroku.



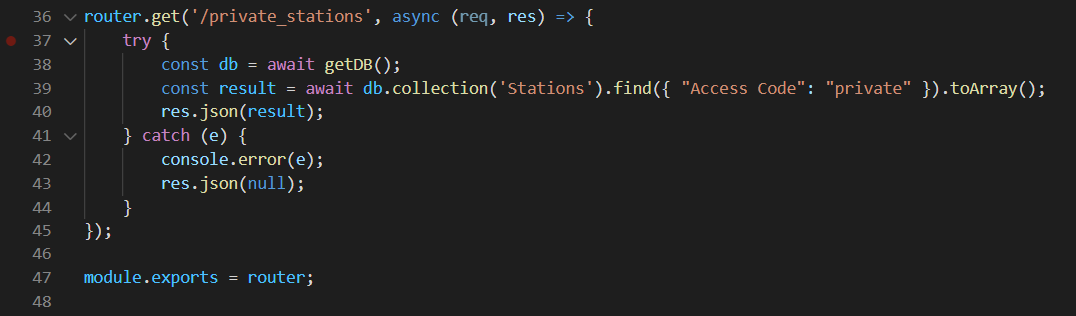
1. para realizar el backend se utiliza node.js, express
2. Creamos las funciones que nos permitirán extraer la información para el menú a implementar.
3. Tenemos primero el método get para las estaciones por ciudades las 24 horas



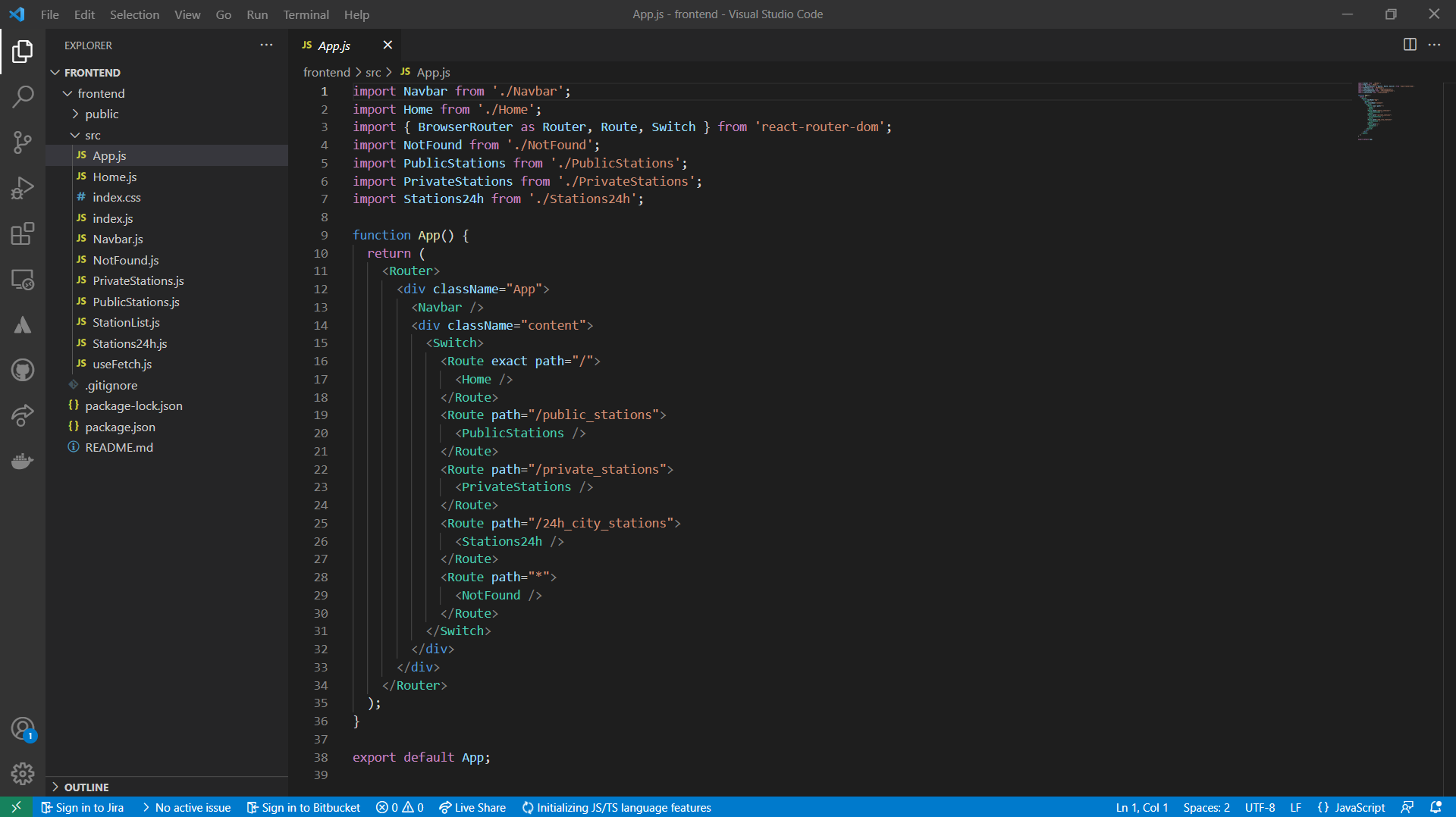
1. El método get para las estaciones públicas



1. Método get para las estaciones privadas



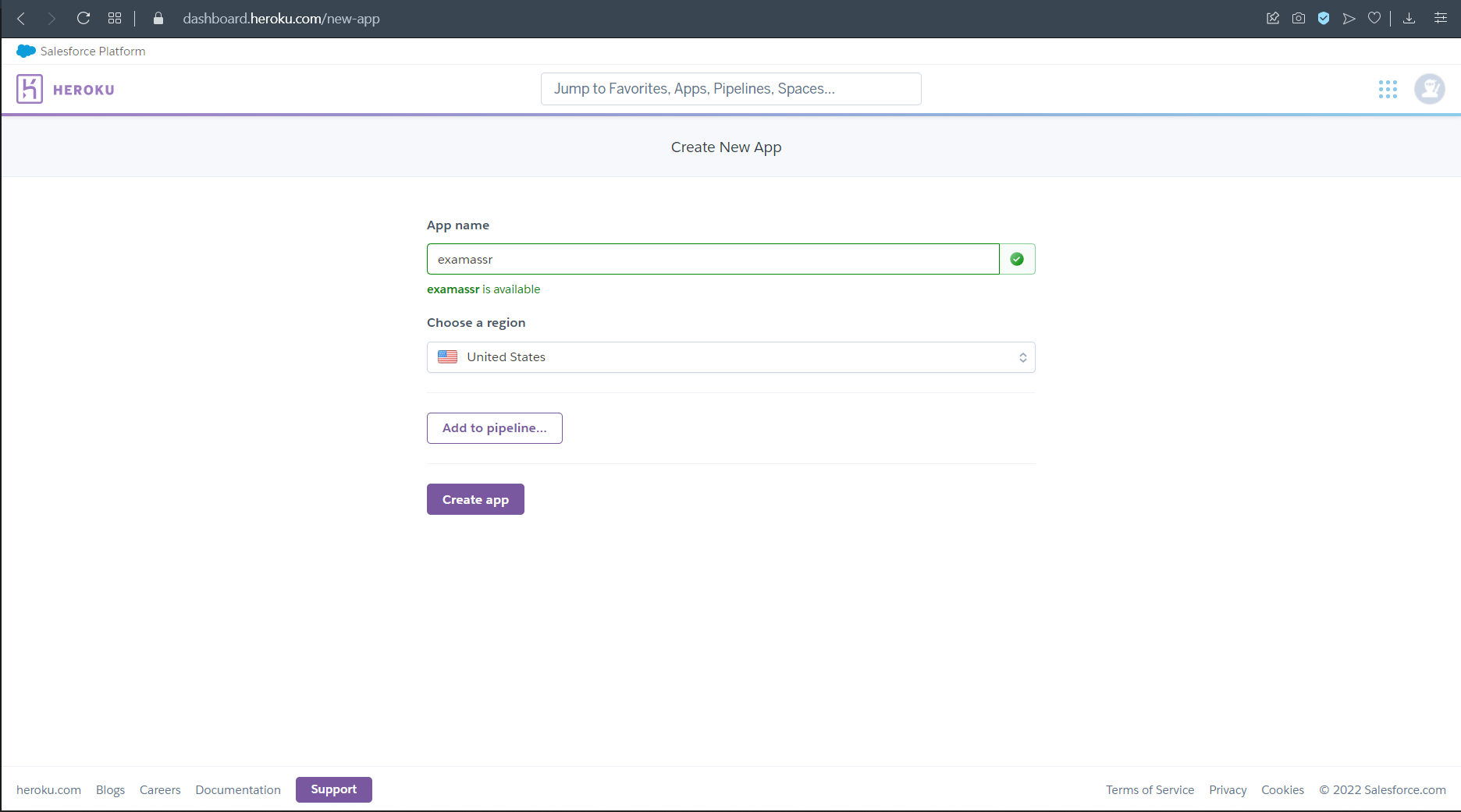
1. Para realizar el **frontend** se solicitó utilizar REACT, además debe tener un menú que nos permite elegir entre las estaciones:
   1. Disponibles por ciudad 24h
   2. Privadas
   3. Públicas
2. Creamos el menú para nuestra aplicación



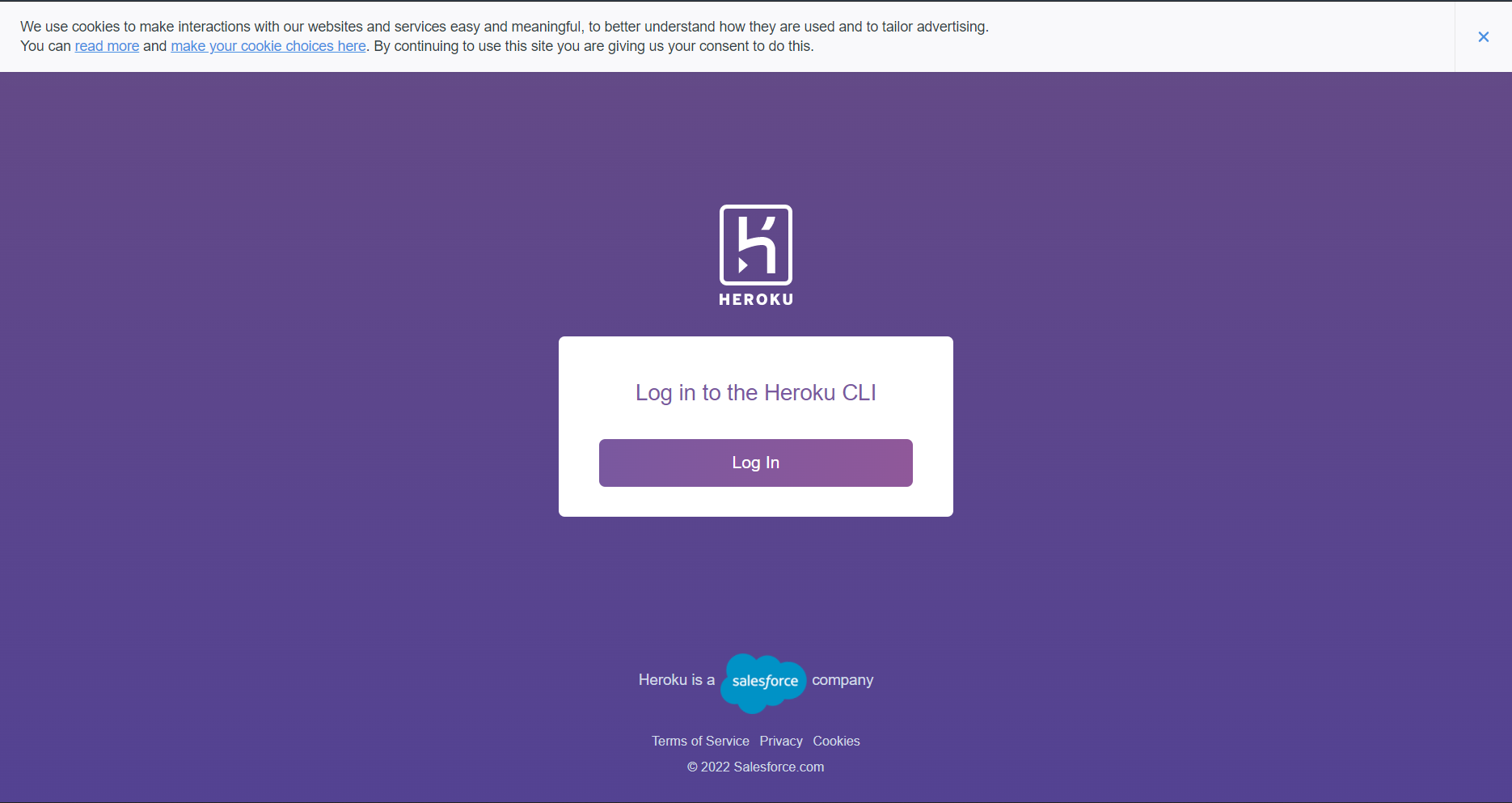
1. El resto de dependencias son creadas para los casos más comunes como error al cargar la página, solicitud inválida o no se reconoce entre otras. El código completo lo puede encontrar en el repositorio de GitHub.

**TEMA 3: DESPLIEGUE EN LA NUBE (30 PUNTOS)**

1. Ahora bien, una vez finalizado el backend y el frontend es hora de desplegar el api en Heroku, para ello nos crearemos una cuenta en <https://www.heroku.com>

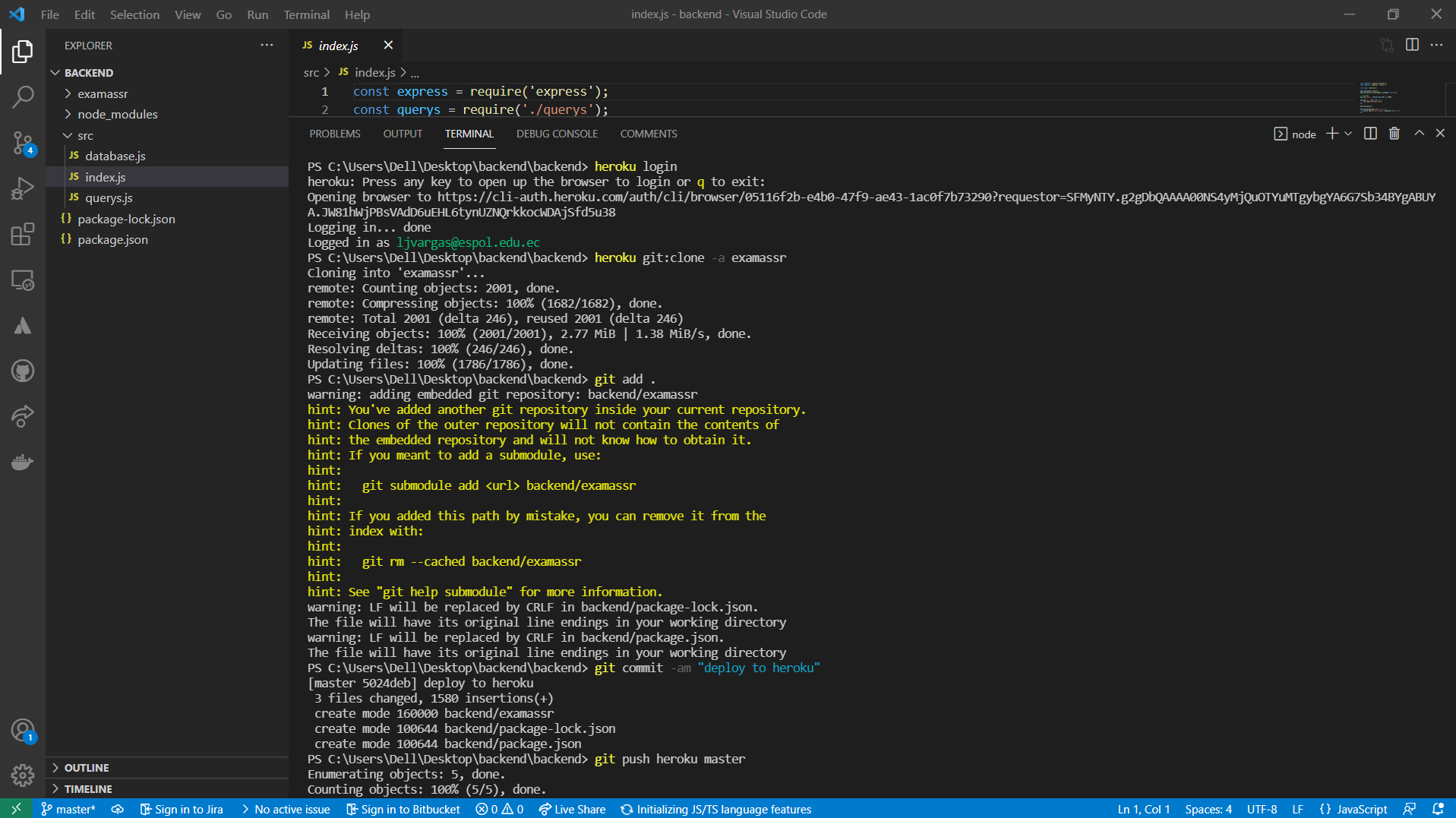


1. Creamos una nueva app, le seteamos los valores como el nombre, ubicación y demás preferencias que tengamos.
2. Cargamos nuetsra api en heroku utilizando:
   1. heroku login, nos va a redirigir a una ventana del navegador en donde nos solicitará iniciar sesión.

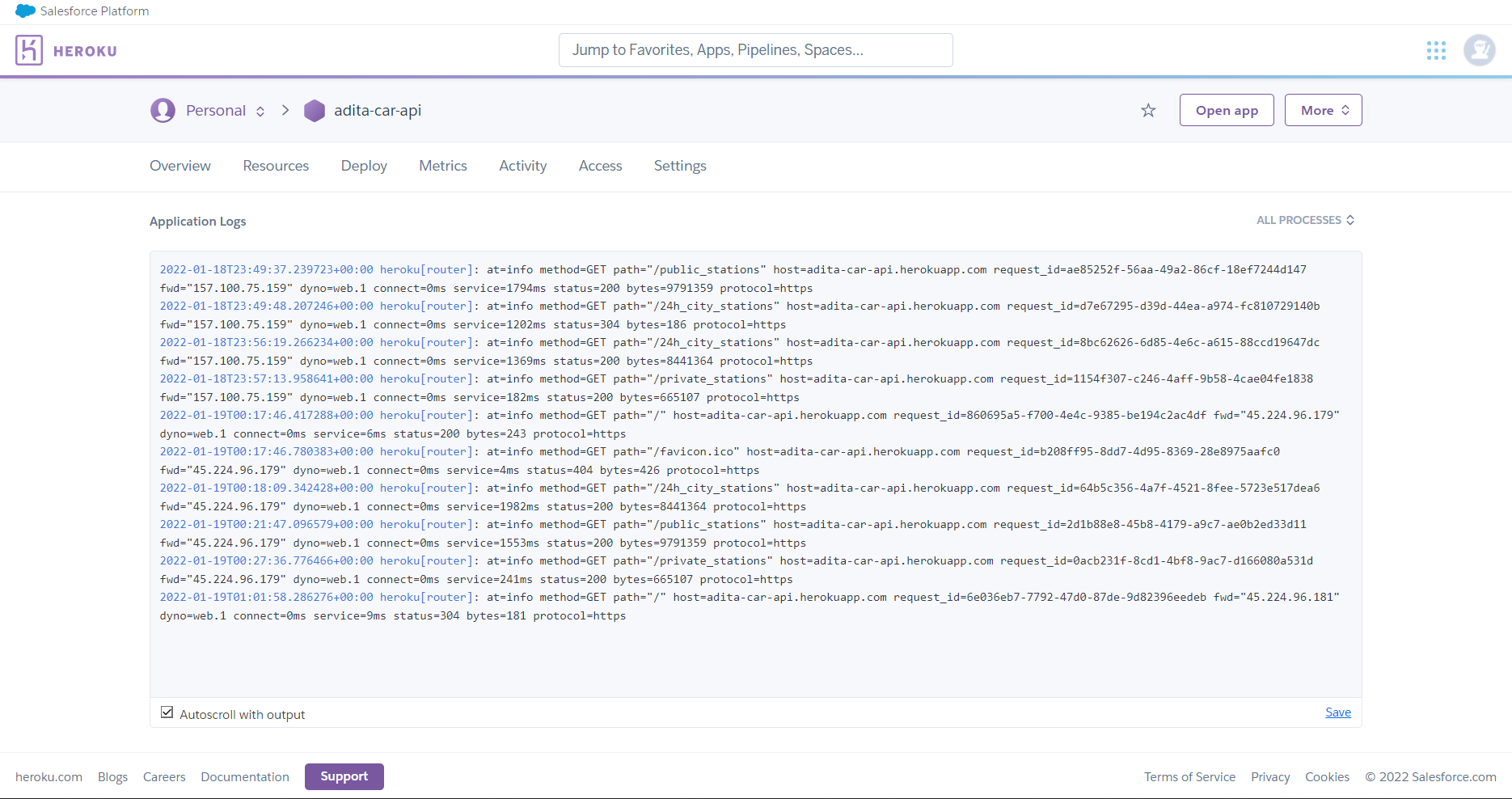


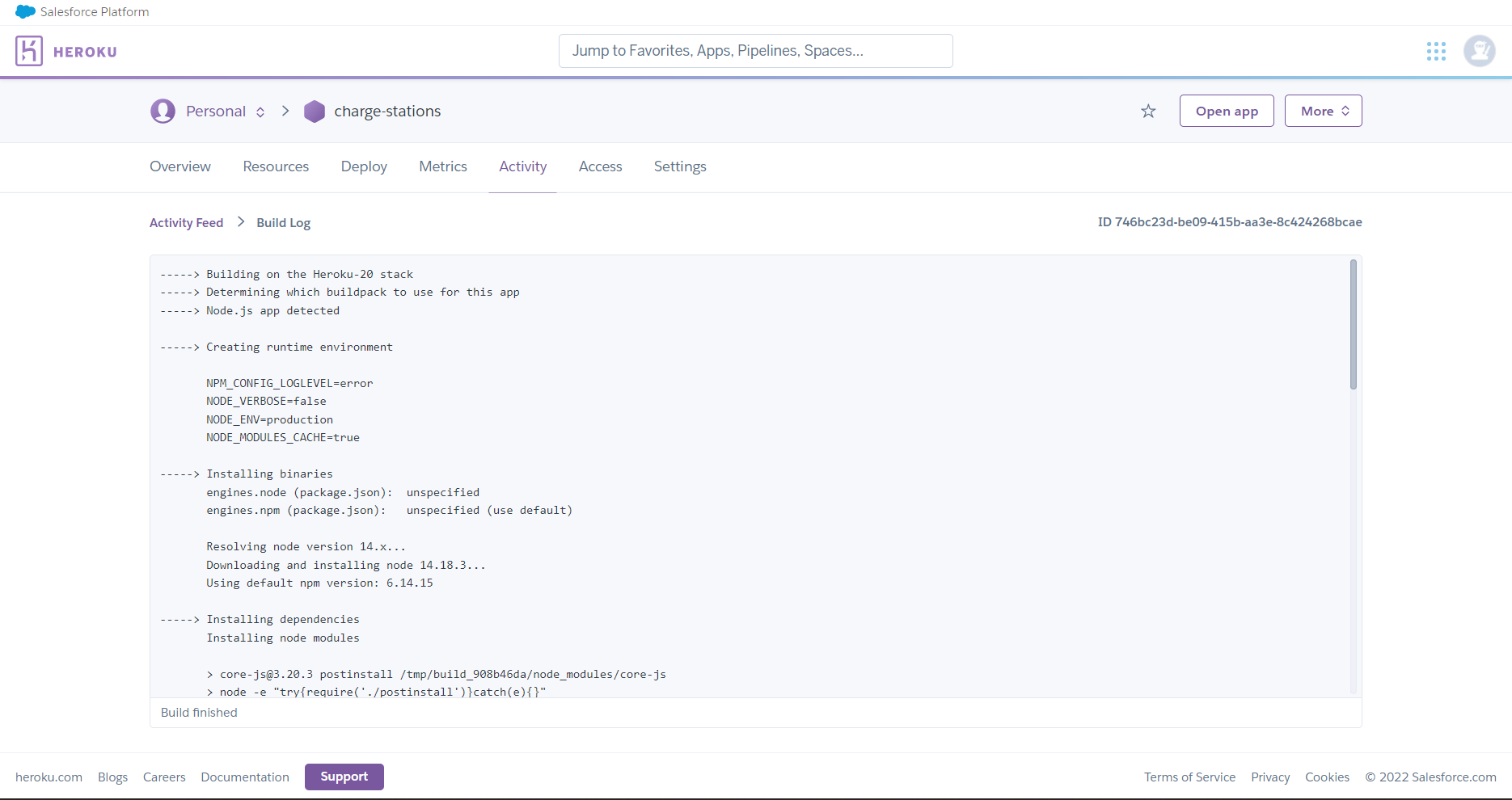
* 1. heroku git:clone -a examassr donde (examassr es el nombre de nuestra app)
  2. git add .
  3. git commit -am "deploy to keroku" (en el Commit podemos indicar el mensaje que necesitemos)
  4. git push heroku master

1. Todo esto lo realizamos desde el terminal de preferencia incorporado en el editor de codigo utilizado (en este caso Visual Studio Code <https://code.visualstudio.com> )

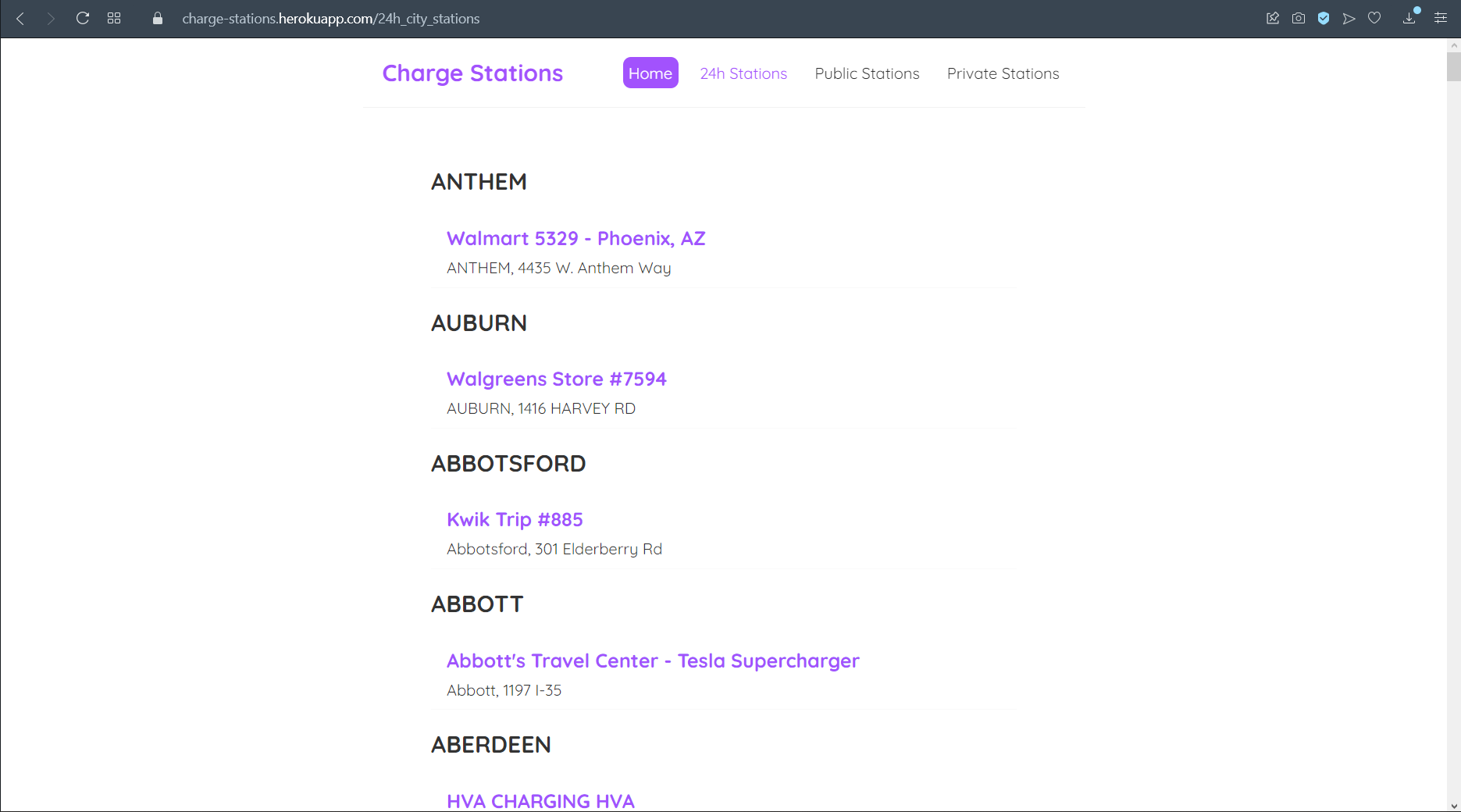


1. Finalmente revisamos que los cambios se reflejen en nuestra cuenta de Heroku

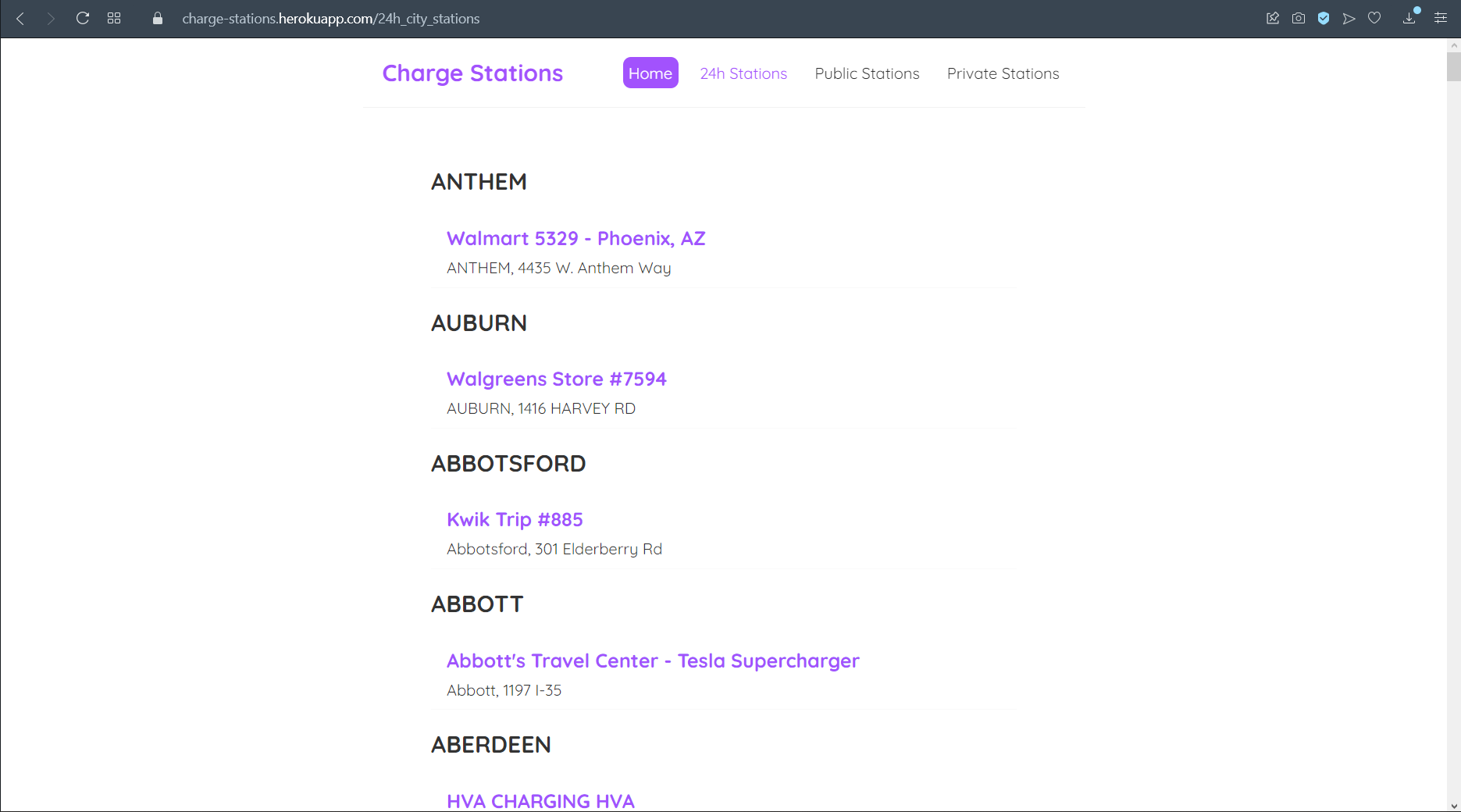




1. Abrimos la aplicación: <https://charge-stations.herokuapp.com> (puede demorar unos segundos en cargar la información)
   1. Caso por ciudad las 24h



* 1. Caso estaciones publicas



* 1. Caso estaciones privadas

