Paso 1. Instalar máquina virtual con Ubuntu. Para en este caso 4GB de RAM y 100GB de almacenamiento.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Paso 2. Instalamos la interfaz gráfica. Url de referencia https://eltallerdelbit.com/interfaz-grafica-en-ubuntu-server/

Sudo apt-get update

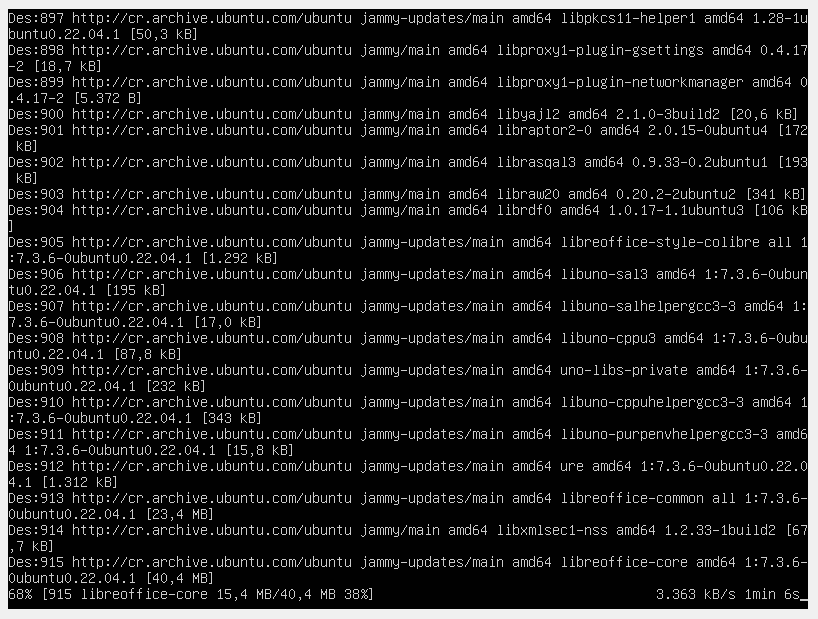
Sudo apt-get upgrade

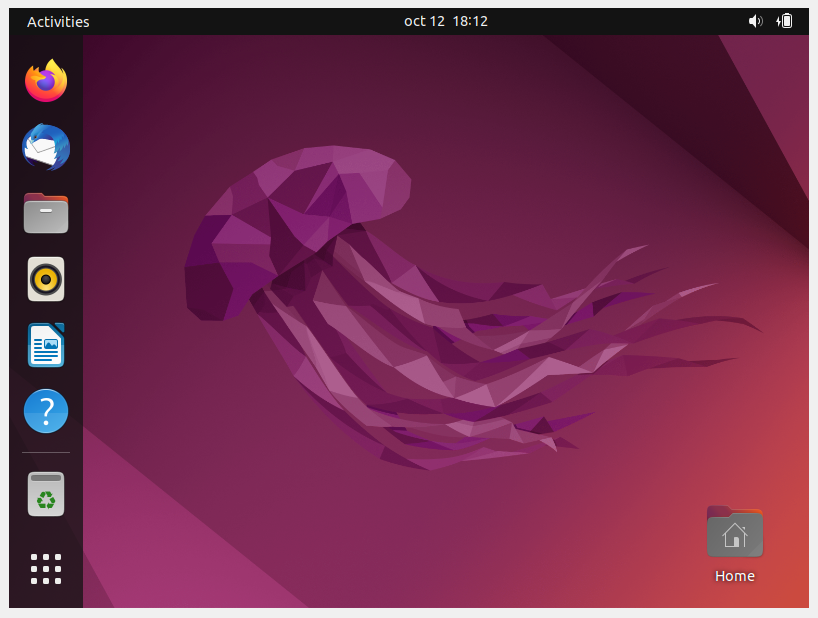
Sudo apt-get install Ubuntu-desktop

Sudo apt-get install gnome-core

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

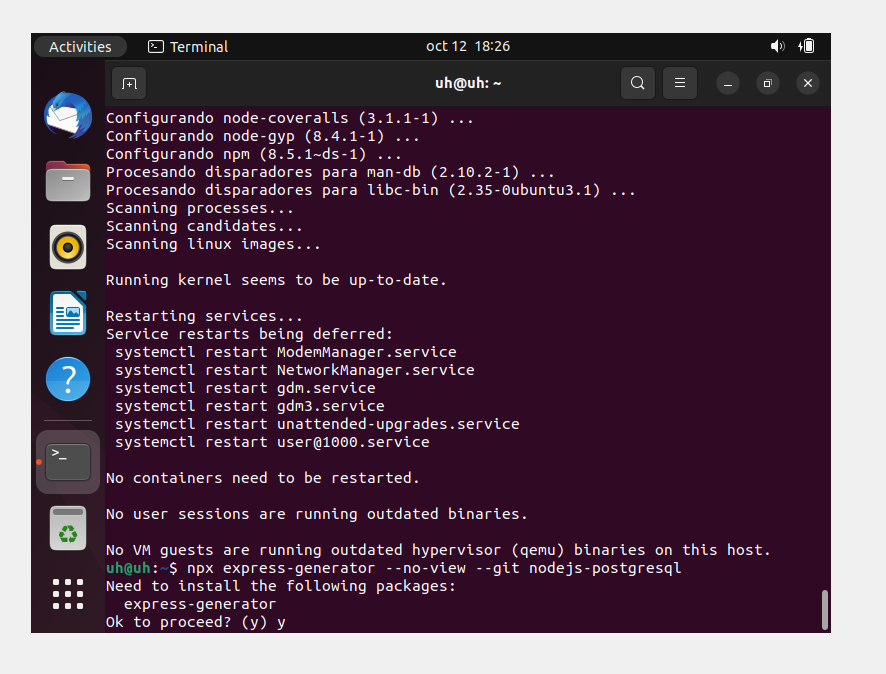




Paso 3. Instalar Node.JS PostgreSQL

Texto

Descripción generada automáticamente



Paso 4. Verificar la instalación con el comando cd nodejs-postgresql && npm install && DEBUG=nodejs-posgresql:\* npm start

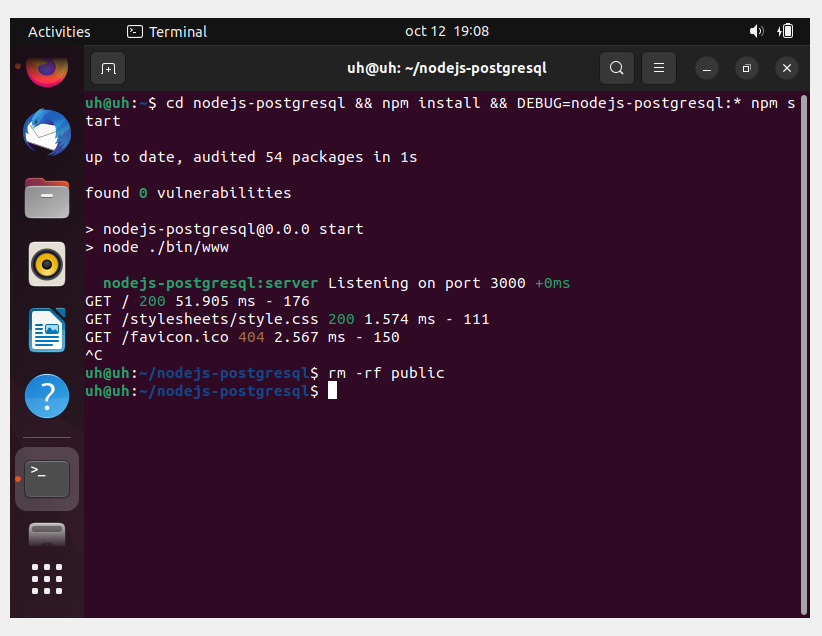
Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Paso 5. Se borra el folder public con rm -rf public

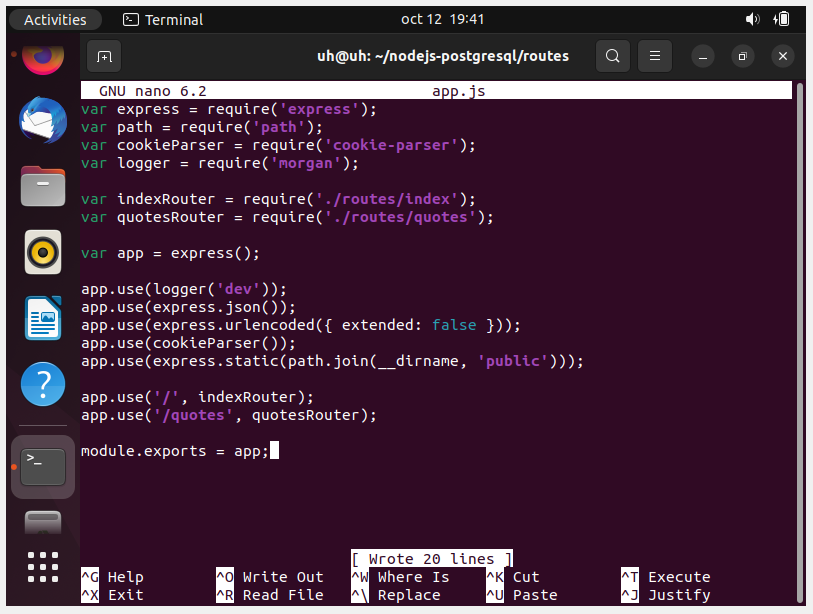


Paso 6. Creamos el archivo quotes.js con el siguiente código usando nano quotes.js

Texto

Descripción generada automáticamente

Paso 7. Modificamos el archivo app.js que está en /home/uh/nodejs-postgresql con el siguiente código:



Paso 8. Modificamos el archivo index.js y lo dejamos así

Texto

Descripción generada automáticamente

Paso 9. Ejecutamos un debug (DEBUG=nodejs-postgresql:\* npm start) y verificamos mediante navegador (localhost:3000/quotes) que este operando el JSON.

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Paso 10. Crear la BD mediante Elephant ingresando al URL <https://elephantsql.com/>

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

Paso 11. En Elephant vamos a browser y copiamos el siguiente Query y ejecutamos

CREATE SEQUENCE quote\_id\_seq START WITH 1 INCREMENT BY 1 NO MINVALUE NO MAXVALUE CACHE 1;

CREATE TABLE quote (

id bigint DEFAULT nextval('quote\_id\_seq'::regclass) NOT NULL PRIMARY KEY,

quote character varying(255) NOT NULL UNIQUE,

author character varying(255) NOT NULL,

created\_at timestamp with time zone DEFAULT CURRENT\_TIMESTAMP NOT NULL,

updated\_at timestamp with time zone DEFAULT CURRENT\_TIMESTAMP NOT NULL

);

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Paso 12. Añadimos en esa misma sección los datos a las tablas con el siguiente Query:

INSERT INTO quote (quote, author) VALUES

('There are only two kinds of languages: the ones people complain about and the ones nobody uses.', 'Bjarne Stroustrup'),

('Any fool can write code that a computer can understand. Good programmers write code that humans can understand.', 'Martin Fowler'),

('First, solve the problem. Then, write the code.', 'John Johnson'),

('Java is to JavaScript what car is to Carpet.', 'Chris Heilmann'),

('Always code as if the guy who ends up maintaining your code will be a violent psychopath who knows where you live.', 'John Woods'),

('I''m not a great programmer; I''m just a good programmer with great habits.', 'Kent Beck'),

('Truth can only be found in one place: the code.', 'Robert C. Martin'),

('If you have to spend effort looking at a fragment of code and figuring out what it''s doing, then you should extract it into a function and name the function after the "what".', 'Martin Fowler'),

('The real problem is that programmers have spent far too much time worrying about efficiency in the wrong places and at the wrong times; premature optimization is the root of all evil (or at least most of it) in programming.', 'Donald Knuth'),

('SQL, Lisp, and Haskell are the only programming languages that I’ve seen where one spends more time thinking than typing.', 'Philip Greenspun'),

('Deleted code is debugged code.', 'Jeff Sickel'),

('There are two ways of constructing a software design: One way is to make it so simple that there are obviously no deficiencies and the other way is to make it so complicated that there are no obvious deficiencies.', 'C.A.R. Hoare'),

('Simplicity is prerequisite for reliability.', 'Edsger W. Dijkstra'),

('There are only two hard things in Computer Science: cache invalidation and naming things.', 'Phil Karlton'),

('Measuring programming progress by lines of code is like measuring aircraft building progress by weight.', 'Bill Gates'),

('Controlling complexity is the essence of computer programming.', 'Brian Kernighan'),

('The only way to learn a new programming language is by writing programs in it.', 'Dennis Ritchie');

Interfaz de usuario gráfica, Texto, Aplicación, Word

Descripción generada automáticamente

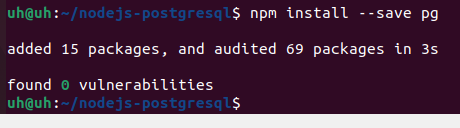
Paso 13. Validamos la inserción de datos ejecutando la siguiente consulta a la BD:

SELECT \* FROM quote;

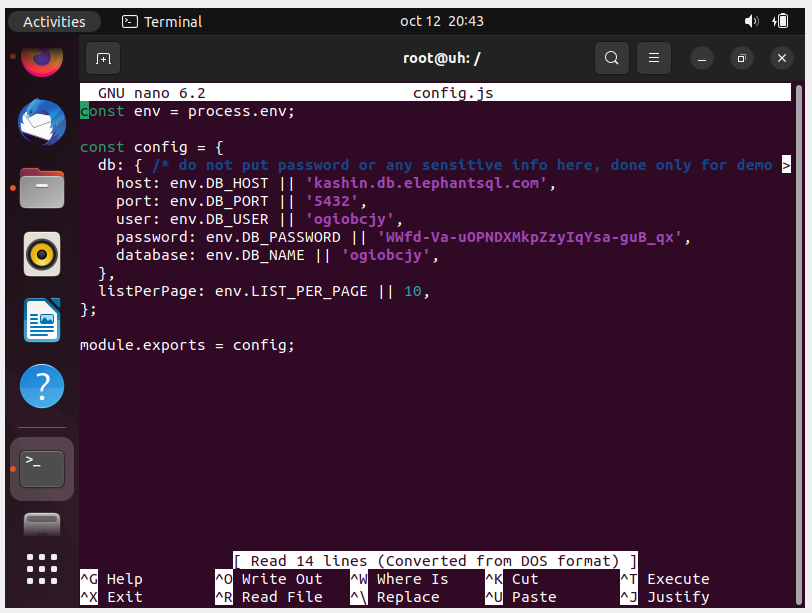
Interfaz de usuario gráfica, Aplicación, Tabla

Descripción generada automáticamente

Paso 14. Ahora sincronizamos la BD con el Node.JS instalando la librería usando npm install –save pg

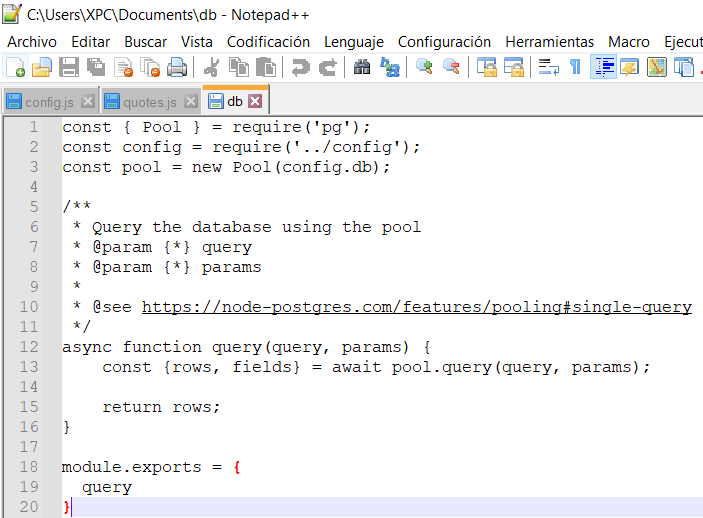


Paso 15. Crear el archivo config.js en root / con el siguiente código cambiando los campos de acceso por los del Elephant SQL.



Paso 16. Crear el directorio services dentro del nodejs-postgresql y colocar los archivos db.js y quotes.js

16.a db.js



16.b quotes.js

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Paso 17. Colocar en root el archivo helper.js



Paso 18. Modificar el archivo routes/quotes.js

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Paso 19. Verificar el acceso a la data de la BD en línea ingresando al navegador web

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Paso 20. Ahora se configura el POST para agregar un nuevo QUOTE modificando el /routes/quotes.js

Texto

Descripción generada automáticamente

Paso 21. Añadimos el método crear en /services/quotes.js

Texto

Descripción generada automáticamente

Paso 22. Añadir un Quote mediante el siguiente comando

Texto

Descripción generada automáticamente