



Pasos para conectarse a vercel



1- Crear cuenta de vercel en Github

Luego de crear la cuenta pasamos visual studio code



2- Instalar Vercel globalmente.

Usando una terminal usamos el comando `npm install -g vercel` para instalar la CLI Línea de comando de vercel.

Cambiamos el nombre del archivo principal del backend a `index.js`

dentro del `package.json` en `"main"`: colocamos `index.js`

```
"main": "index.js",
```


y en `"scripts"`: colocamos `"server": "node index.js"`

```
"scripts": {  
  "server": "node index.js"  
},
```



3- Crear archivo vercel.json

```
{
  "version": 2,
  "builds": [
    {
      "src": "index.js",
      "use": "@vercel/node"
    }
  ],
  "routes": [
    {
      "src": "/(.*)",
      "dest": "index.js"
    }
  ]
}
```



4- Iniciar vercel usando modo desarrollador vercel dev

Usando una terminal dentro de la carpeta [nombre de la carpeta backend]

usamos el comando vercel dev

nos va a iniciar la configuración para subir el proyecto

- Set up and develop colocamos **y/enter**
- Si nos sale el nombre de usuario que colocamos en vercel o en git **y/enter**
- Project exist **n/enter**
- En la ruta de directorio nos va a salir la principal **./** le damos **enter**
- Al terminar no va a devolver un URL usando el **localhost:3000** control+click o alt+click para abrir desde consola de VSCODE



5- Index.js

Comentamos o eliminamos el código que levanta el index.js que usamos anteriormente de forma local.

```
app.listen(PORT, function() {})  
app.listen(PORT, () => {})
```

y al final del archivo colocamos `module.exports=app` para que vercel por detrás use el `app` de express



6- Verificamos conexión con vercel

Usando la ruta que nos brindó vercel en la terminal como dev `localhost:3000`

en pantalla deberíamos de tener `can't get ./`

si es así se conectó correctamente, si queremos agregar un mensaje más detallado, si está conectado, podemos colocar un mensaje a la ruta principal

```
app.get("/", (req, res) => {  
  res.send("Api Funcionando");  
});
```

y si volvemos a la ruta otra vez en el navegador nos debería mostrar

`API Funcionando`



7- Probar si obtiene las tarjetas

Cambiamos la ruta en el `play service` por la ruta que nos dio vercel.

Si obtiene las cartas podemos subir los cambio a GitHub y luego subir los archivos a producción para que vercel no brinde el `URL` oficial de backend.

Usando el comando `vercel --prod` subirá los cambios a vercel, todo cambio dentro de la carpeta backend debemos de subirlos a GitHub y de igual forma a vercel usando el comando `vercel --prod` siempre nos devolverá la misma URL