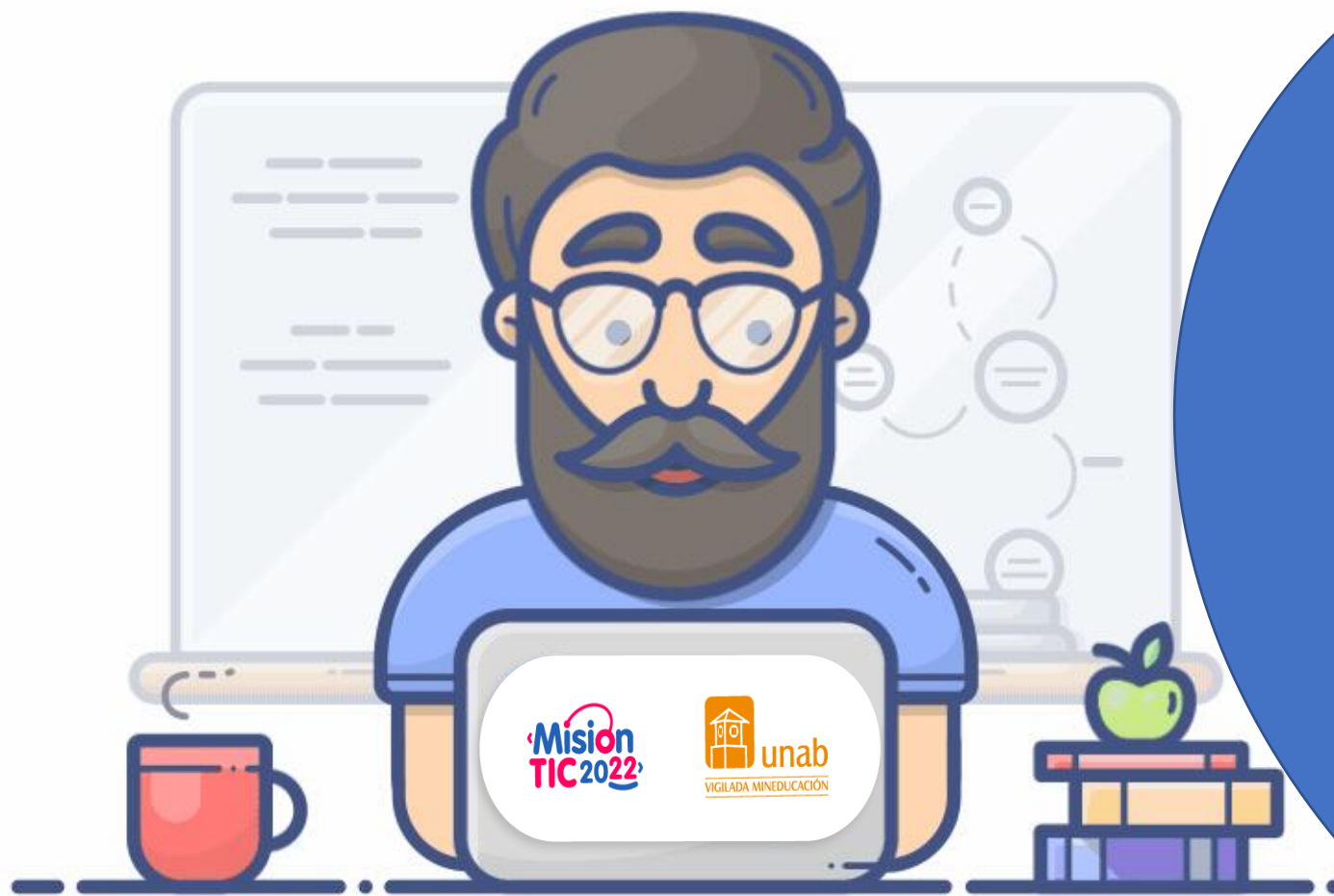




Recuerda marcar tu asistencia



Asistencia en LMS



Semana 2

Servidor en Node JS

Ing Edwin Garcia



El futuro digital
es de todos

MinTIC

Hechos

QUE

CONECTAN

Misión
TIC 2022

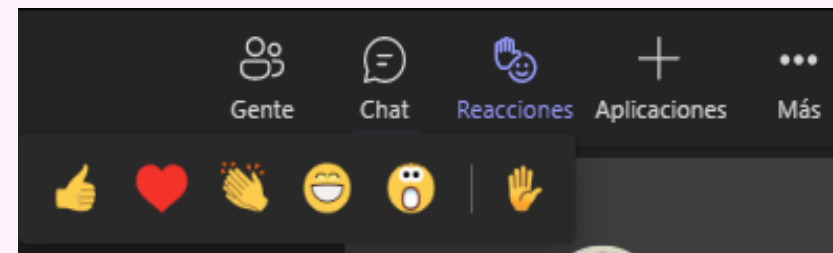
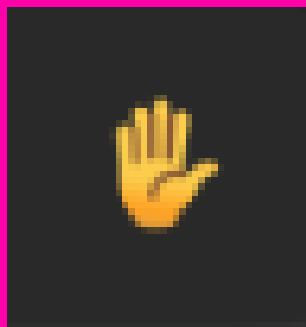
unab
VIGILADA MINEDUCACIÓN



Recuerden marcar su asistencia

Asistencia en LMS

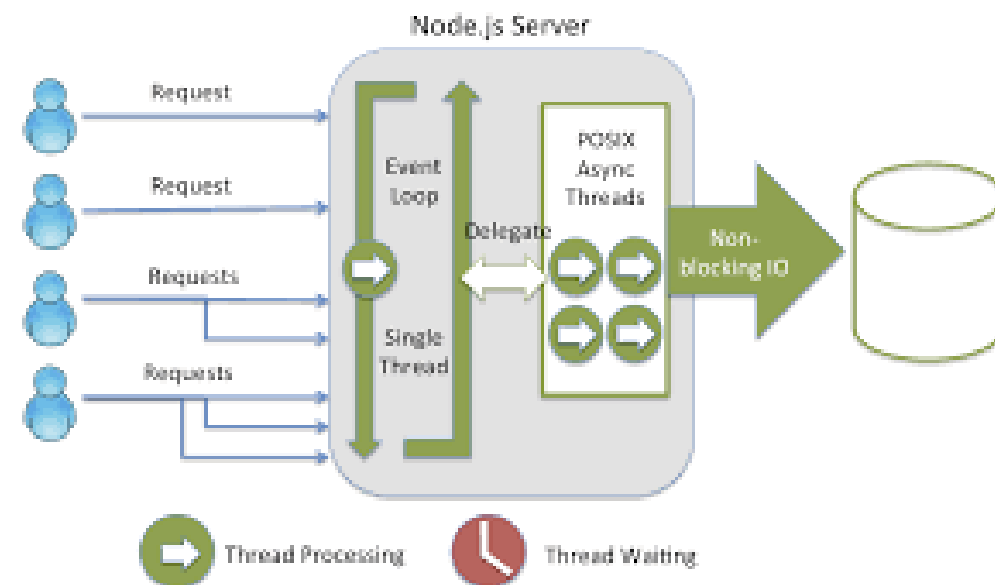
**Pedir la
palabra para
realizar una
pregunta o
una
intervención**





Servidores con Node JS

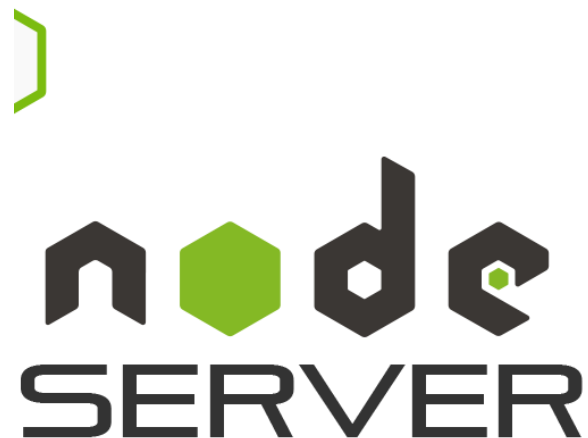
Aplicando la versatilidad y eficiencia de Javascript, con node JS se logra aprovechar esta ventaja para crear sistemas servidor que aprovechar mejor los recursos.





Http

Es un modulo que permite implementar un sistema cliente servidor, permitiendo recibir y procesar solicitudes.



```
C:\Users\LENOVO>netstat -an
```

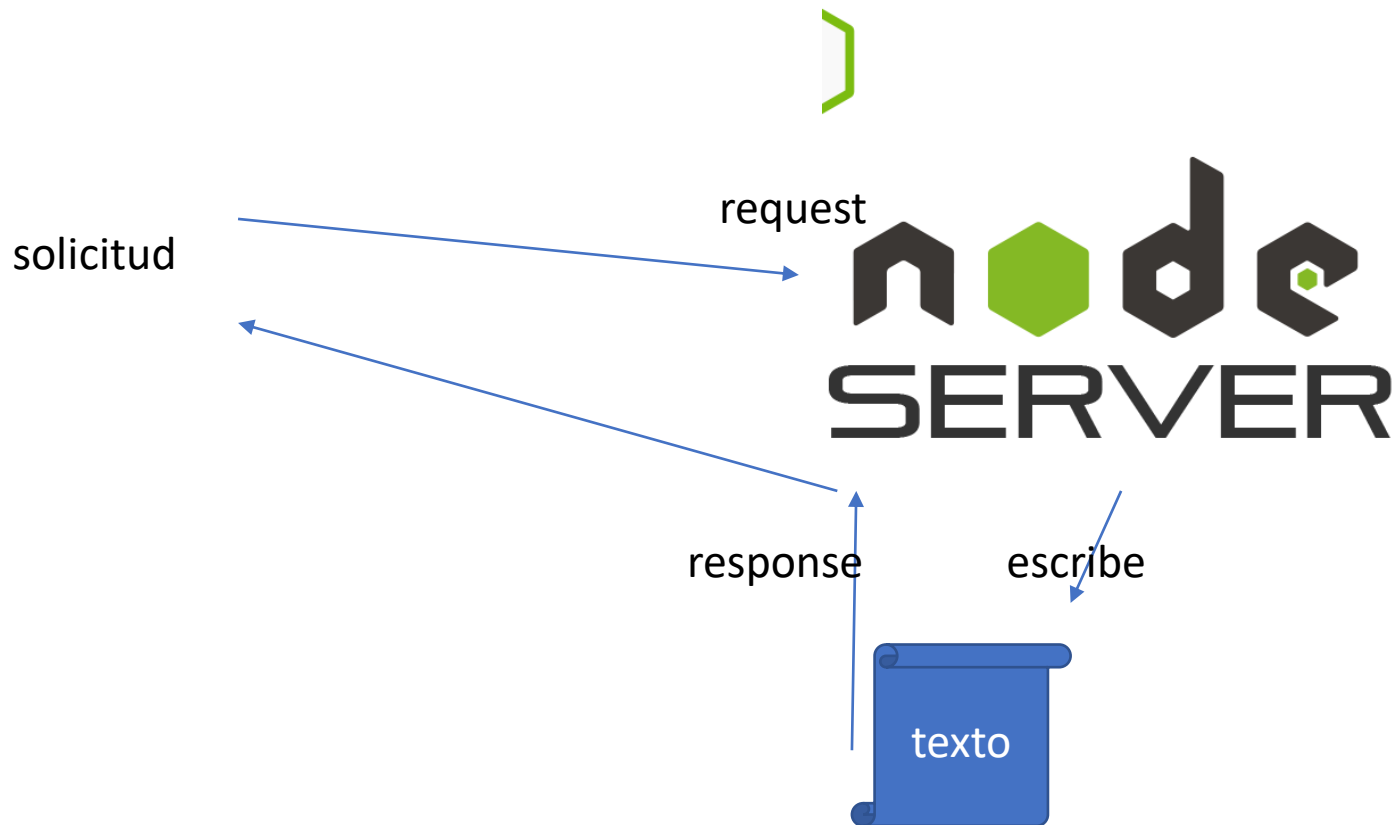
TCP		UDP	
FTP	20,21	DNS	53
SSH	22	BooTPS/DHCP	67
Telnet	23	TFTP	69
SMTP	25	NTP	123
DNS	53	SNMP	161
HTTP	80		
POP3	110		
IMAP4	143		
HTTPS	443		



```
const http = require('http')

const puerto = 3000
http.createServer(function(request, response){
    response.write("hola mundo")
    response.end()
}).listen(puerto)

console.log("servidor activo en el puerto: " + puerto)
```



```
const http = require('http')

const puerto = 3000
http.createServer((request, response) => {
  response.write("hola mundo")
  response.end()
}).listen(puerto)

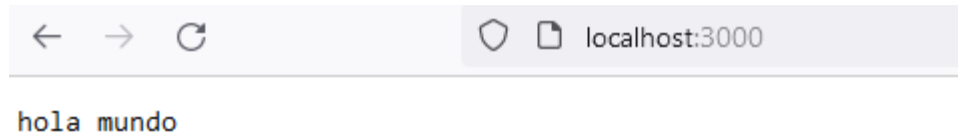
console.log("servidor activo en el puerto: " + puerto)
```



Direccionamiento de peticiones

- Unab.co
- unab.co/inicio
- Unab.co/registro
- Unab.co/misiontic2022

request => podemos conocer que solicitudes se realizan.



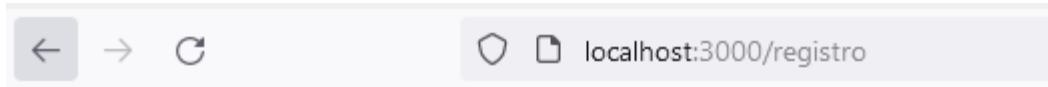
????



```
const puerto = 3000
http.createServer((request, response) => {
  console.log(request.url)
  response.write("hola mundo")
  response.end()
}).listen(puerto)

console.log("servidor activo en el puerto: " + puerto)
```

```
C:\Users\LENOVO\Desktop\node\G83>node index
servidor activo en el puerto: 3000
/
█
```

hola mundo

```
C:\Users\LENOVO\Desktop\node\G83>node index  
servidor activo en el puerto: 3000  
/  
/registro
```



De esta forma podemos direccionar solicitudes recibidas mediante la captura de la variable request.



Desarrollo de Aplicaciones Web



El futuro digital
es de todos

MinTIC

```
console.log(request.url)
if(request.url == '/registro'){
  console.log('peticion registro')
}
```

localhost:3000/registro

```
C:\Users\LENOVO\Desktop\node\G83>node index
servidor activo en el puerto: 3000
/registro
peticion registro
/favicon.ico
```

Mision
TIC 2022



Hechos
QUE CONECTAN ✓



```
console.log(request.url)
if(request.url == '/registro'){
    console.log('peticion registro')
    response.write('pagina de registro')
}
```



localhost:3000/registro

pagina de registrohola mundo



```
console.log('peticion registro')  
response.write('pagina de registro')  
return response.end()  
}
```



localhost:3000/registro

pagina de registro



```
http.createServer((request, response) => {  
  console.log(request.url)  
  if(request.url == '/'){  
    response.write("Pagina de Inicio (Index) ")  
    return response.end()  
  }  
  
  if(request.url == '/registro'){  
    console.log('peticion registro')  
    response.write('pagina de registro')  
    return response.end()  
  }  
}) .listen(puerto)
```

Dando formato html

```
}  
  
if(request.url == '/registro'){  
    console.log('peticion regis  
    response.write('pagina de r  
    return response.end()  
}  
  
response.write(`  
    <html>  
    <h1>No encontrado</h1>  
    </html>`)  
response.end()
```

localhost:3000/otro

localhost:3000/otro

No encontrado



```
ers\LENOVO\Desktop\node\G83I> npm init  
ity will walk you through creating a package
```

```
package name: (g83i)  
version: (1.0.0)  
description: test  
entry point: (index.js)
```

```
{  
  "name": "g83i",  
  "version": "1.0.0",  
  "description": "test",  
  "main": "index.js",  
  "scripts": {  
    "test": "echo \"Error: no test specified\" && exit 1"  
  },  
  "keywords": [  
    "marcadores",  
    "equipos",  
    "deportes"  
  ],  
  "author": "edwin g",  
  "license": "ISC"  
}
```

```
Is this OK? (yes)
```



Desarrollo de Aplicaciones Web



El futuro digital
es de todos

MinTIC

▼ G83I

{ } package.json

P

Mision
TIC 2022



Hechos
QUE CONECTAN ✓



```
  ▶ Debug
  "scripts": {
    "ejecutar" : "node index.js",
    "test": "echo \"Error: no test specified\" && exit 1"
  },
```

```
PS C:\Users\LENOVO\Desktop\node\G83I> npm run ejecutar

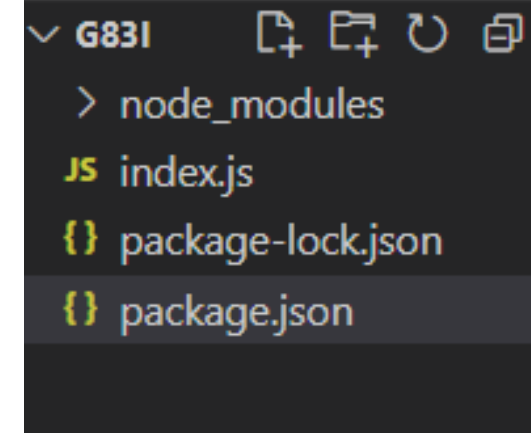
> g83i@1.0.0 ejecutar
> node index.js

hola mundo
PS C:\Users\LENOVO\Desktop\node\G83I>
```




Instalar módulos

```
PS C:\Users\LENOVO\Desktop\node\G83I> npm i colors  
  
added 1 package, and audited 2 packages in 891ms  
  
found 0 vulnerabilities
```





Desarrollo de Aplicaciones Web



El futuro digital
es de todos

MinTIC

```
index.js 7 ...
1  const c = require('colors')
2  console.log("hola mundo".bgCyan)
3  console.log("misiontic 2022".bgRed)
4  console.log("G83".bgYellow)
5
```

Rainbow
Color de texto

```
PS C:\Users\LENOVO\Desktop\node\G831> npm run ejecutar
```

```
> g83i@1.0.0 ejecutar
```

```
> node index.js
```

```
hola mundo
```

```
misiontic 2022
```

```
G83
```





```
G83I> npm start
```



Detectar cambios y ejecutar automáticamente

```
PS C:\Users\LENOVO\Desktop\node\G83I> npm i nodemon  
  
added 32 packages, and audited 34 packages in 3s  
  
3 packages are looking for funding  
  run `npm fund` for details  
  
found 0 vulnerabilities
```

```
"main": "index.js",  
  "scripts": {  
    "start": "nodemon index.js",  
    "test": "echo \\\"Error: no test specified\\\" && exit 1"  
  },
```



Si presenta error “el termino nodemon no se reconoce”

```
npm install -g nodemon
```




index.js > ...

```
1  const c = require('colors')
2  console.log("hola mundo".bgCyan)
3  console.log("misiontic 2022".bgRed)
4  console.log("Grupo G83".bgYellow)
5  |
```

```
[nodemon] starting `node index.js`
hola mundo
misiontic 2022
Grupo G83
[nodemon] clean exit - waiting for changes before restart
[nodemon] restarting due to changes...
[nodemon] starting `node index.js`
hola mundo
misiontic 2022
G83
[nodemon] clean exit - waiting for changes before restart
```

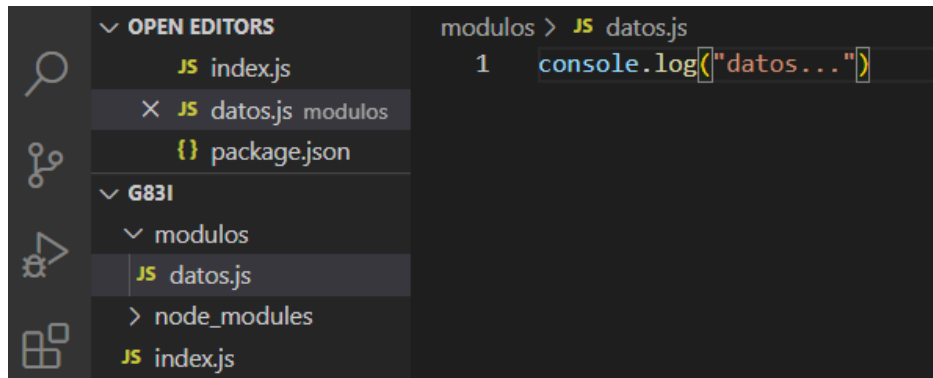


Desarrollo de Aplicaciones Web



El futuro digital
es de todos

MinTIC



```
JS index.js > ...  
1  const c = require('colors')  
2  
3  const f = require('./modulos/datos')  
4  
5  console.log("hola mundo".bgCyan)  
6  console.log("misiontic 2022".bgRed)
```

```
[nodemon] clean exit - waiting for changes before restart  
[nodemon] restarting due to changes...  
[nodemon] starting `node index.js`  
datos...  
hola mundo  
misiontic 2022  
Grupo G83  
[nodemon] clean exit - waiting for changes before restart
```

Mision
TIC 2022



echos

QUE CONECTAN



Al cerrar

```
[nodemon] clean exit - waiting for changes before  
¿Desea terminar el trabajo por lotes (S/N)?
```



- Eliminar un modulo remove



Desarrollo de Aplicaciones Web



El futuro digital
es de todos

MinTIC

```

  ✓ OPEN EDITORS 1 unsaved
    JS index.js
    • <> registro.html mo...
    JS datos.js  modulos
    {} package.json
  ✓ G83I
    ✓ modulos
      JS datos.js
      <> registro.html
      > node_modules

modulos > <> registro.html > ...
1  <h1>hola mundo</h1>
2

```



Hechos
QUE CONECTAN ✓



```
8
9  const fs = require('fs')
10 const puerto = 3000
11 const http = require('http')
12
13 http.createServer((req,res)=>{
14   console.log(req.url)
15   if(req.url == '/registro'){
16     const arch = fs.readFileSync('./modulos/registro.html' , 'utf-8')
17     res.write(arch)
18     return res.end()
19   }
20 }).listen(puerto)
```



```
OPEN EDITORS
JS index.js
X <> registro.html mo...
JS datos.js modulos
{} package.json
5831
✓ modulos
JS datos.js
<> registro.html
> node_modules

modulos > <> registro.html > ul
1 <h1>hola mundo</h1>
2 <ul>
3   <li>Equipo 1</li>
4   <li>Equipo 2</li>
5   <li>Equipo 3</li>
6 </ul>
```



localhost:3000/registro

hola mundo

- Equipo 1
- Equipo 2
- Equipo 3





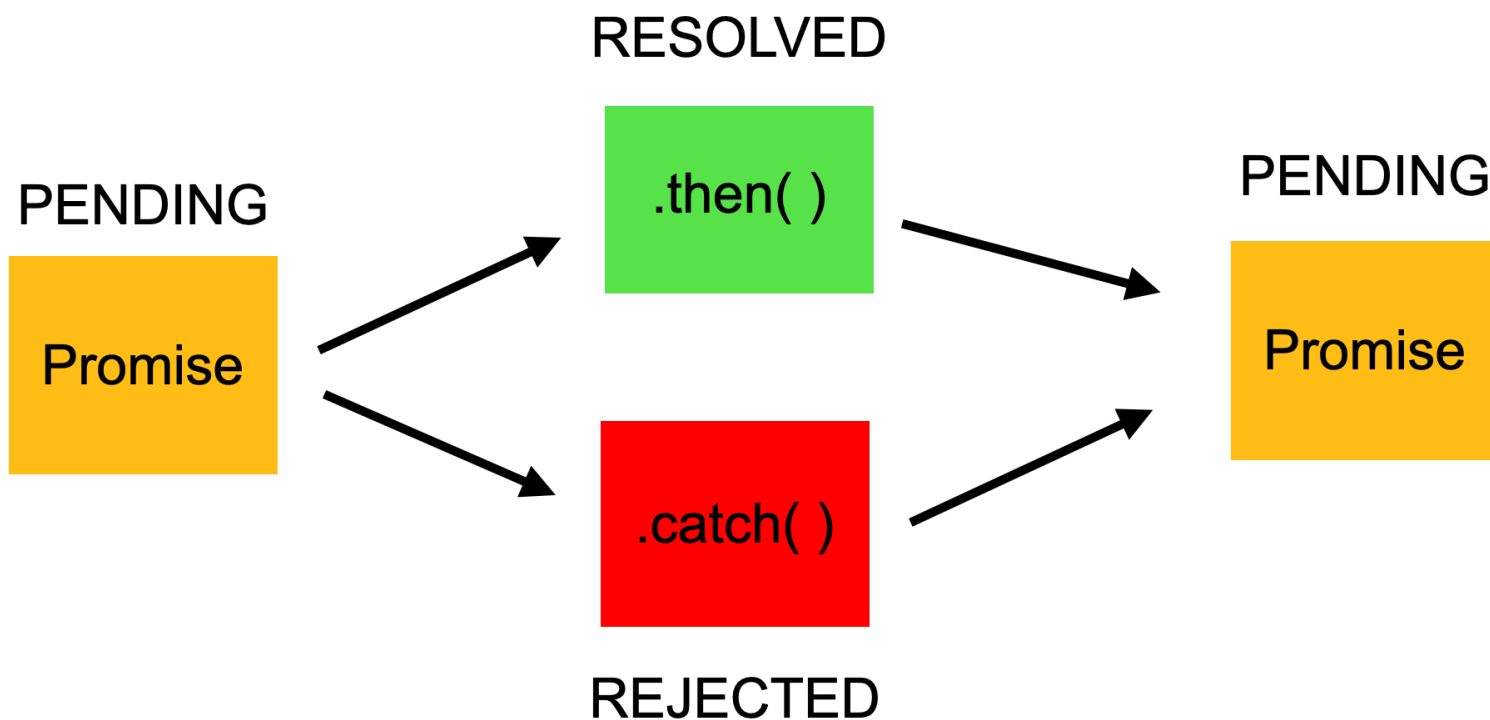
Agregamos el cdn de bootstrap



Resolviendo promesas

Una [Promise](#) (promesa en castellano) es un objeto que representa la terminación o el fracaso de una operación asíncrona. Dado que la mayoría de las personas consumen promises ya creadas, esta guía explicará primero cómo consumirlas, y luego cómo crearlas.

Esencialmente, una promesa es un objeto devuelto al cual se adjuntan funciones callback, en lugar de pasar callbacks a una función.





```
const { readFile } = require('fs')

function getText(url){
  return new Promise(function (resolve,reject){
    readFile(url,'utf-8', (err,data) =>{
      if(err){
        reject(err)
      }
      resolve(data)
    })
  })
}
```

```
getText('./modulos/registro.html')
  .then((result)=>{
    console.log(result)
  })
  .catch((error)=>{ console.log(error)})
```



Asincronia

Tenemos dos partes cuando usamos `async/await` en nuestro código.

En primer lugar, tenemos la palabra clave `async`, que se pone delante de una declaración de función para convertirla en una función `async`.



Asincronia

Una función asíncrona es una función que sabe que es posible que se use la palabra clave `await` dentro de ella para invocar código asíncrono.





La palabra clave `async` se añade a las funciones para que devuelvan una promesa en lugar de un valor directamente.



```
3  
4  async function cargar(){  
5      const res = await getText('./modulos/registro.html')  
6      console.log(res)  
7  }  
8  
9  
0  cargar()
```




Estructura directorios

- config
- controllers
- middleware
- models
- node_modules
- routes
-  index
-  package.json
-  package-lock.json
-  variables.env



- https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Using_promises
- <https://proyectosbeta.net/2021/04/estructura-basica-de-un-proyecto-web-con-nodejs-back-end/>
- <https://www.youtube.com/watch?v=i3OdKwuBjeM>
- https://www.google.com/search?q=node+js+promise+espa%C3%B1ol&client=firefox-b-d&sxsr=ALiCzsbQHgs73bzHGZMzFgWBQCVUMfecYw:1663065993753&source=lnms&tbm=isch&sa=X&ved=2ahUKEwjlvK7Oy5H6AhWOTTABHaaTBpYQ_AUoA3oECAEQBQ&biw=1366&bih=643&dpr=1#imgsrc=AdBu13QBjuuU7M

Bibliografía





El futuro digital
es de todos

MinTIC

**Practicar los
temas tratados**

Mision
TIC2022



Hechos
QUE CONECTAN ✓

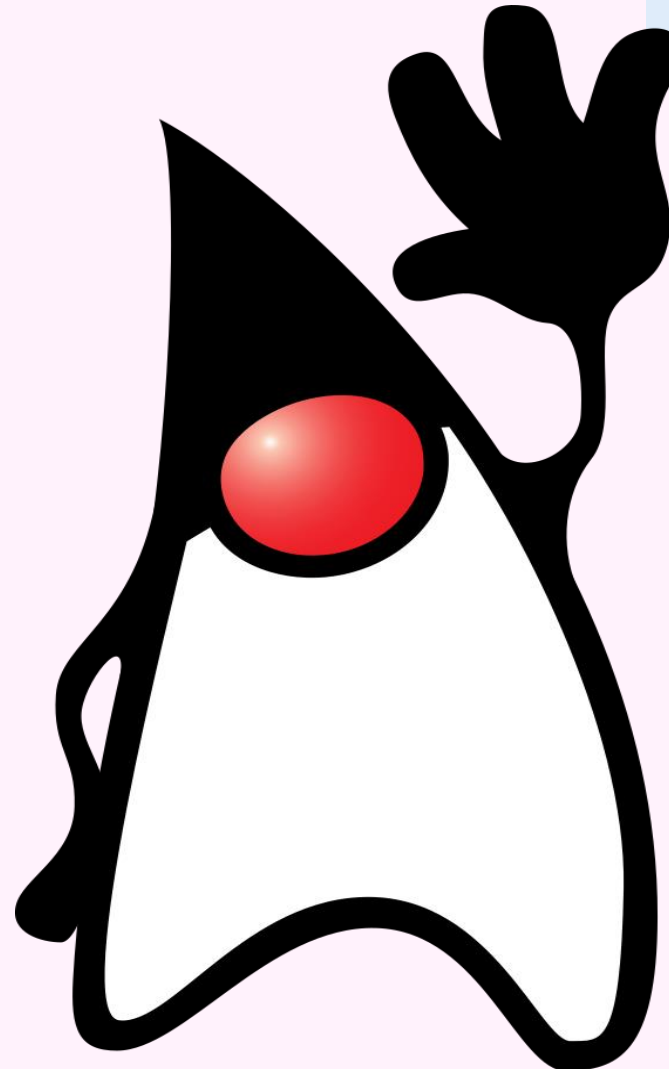
Nos vemos en la siguiente sesión



**El futuro digital
es de todos**

MinTIC

**Muchas
Gracias**



www.mintic.gov.co

**Misión
TIC2022**

unab
VIGILADA MINEDUCACIÓN