



unl

Universidad
Nacional
de Loja

POSGRADO

Desarrollo de Software en Ambientes Cloud

Maestría en Ingeniería en Software - UNL



Carrera de Ingeniería en
Sistemas / Computación



unl

Universidad
Nacional
de Loja





Bienvenidos!

Ing. Roberth Figueroa Díaz M.Sc.

roberth.figueroa@unl.edu.ec

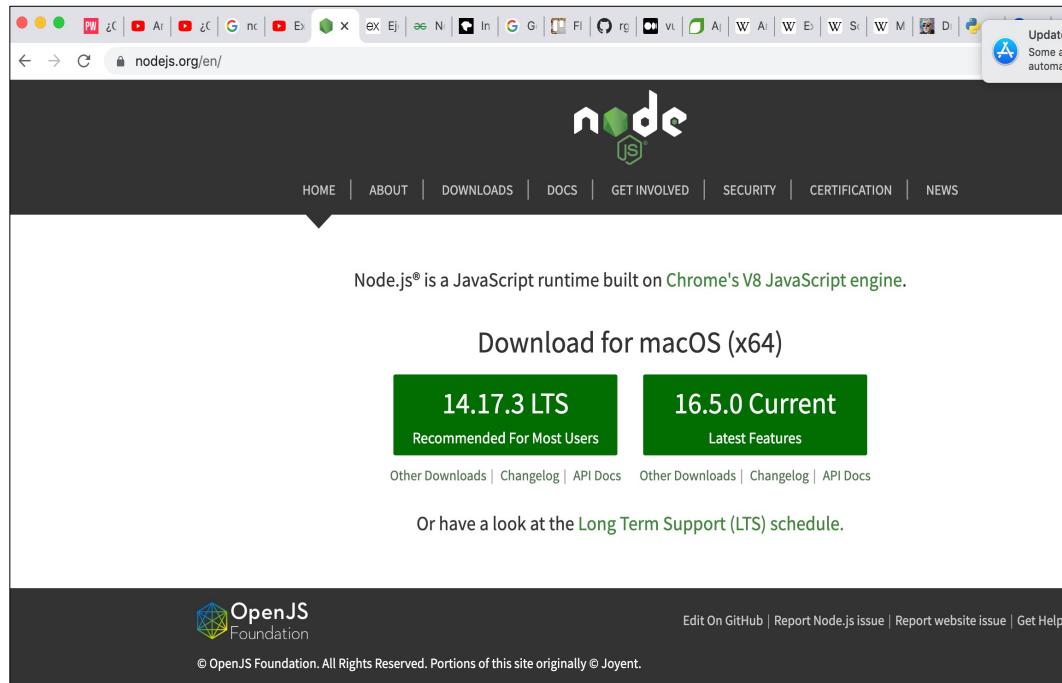
Profesor



Agenda

1. Tecnologías del lado del Servidor o Backend
2. Introducción general a NodoJS
3. Instalando Framework Node.js
 - a) Usando Express
 - b) Usando MongoDB

Node.js



The screenshot shows the official Node.js website at nodejs.org/en/. The page features a dark header with the Node.js logo and navigation links for HOME, ABOUT, DOWNLOADS, DOCS, GET INVOLVED, SECURITY, CERTIFICATION, and NEWS. Below the header, a green banner states "Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine." Two download buttons are prominently displayed: a green one for "14.17.3 LTS" (Recommended For Most Users) and a white one for "16.5.0 Current" (Latest Features). At the bottom, there are links for "Other Downloads | Changelog | API Docs" and "Edit On GitHub | Report Node.js issue | Report website issue | Get Help". The footer also includes a copyright notice: "© OpenJS Foundation. All Rights Reserved. Portions of this site originally © Joyent."

<https://nodejs.org/> → Framework Node.js

Node.js

- Framework de lado del Backend multiplataforma soporta Javascript debido a que utiliza el motor de ejecución V8 de Google
- Node tiene una API extensa de utilitarios que facilita la organización, funcionalidades, middleware(lógica de intercambio) y routing (enrutamiento), objetos Http, renderizado dinámico Html entre otros.

npm → Node Package Manager

<https://www.npmjs.com/>

Instalando Node.js

En nuestro caso se seguirá el proceso en el sistema operativo *MacOS*:

1. Utilizar el comando: → **npm install -g n**
2. Ejecutar el comando: → **n lts**
3. Ejecutar el comando: **node --version** o **node -v** para visualizar la versión instalada

```
[macrob:projects roberth$ node -vers
node: bad option: -version
[macrob:projects]roberth$ node -v12.
v14.17.3 [Screenshot 2021-07-19 at 12.0
macrob:projects roberth$ ] [Screenshot 2021-07-19 at 11.5]
```

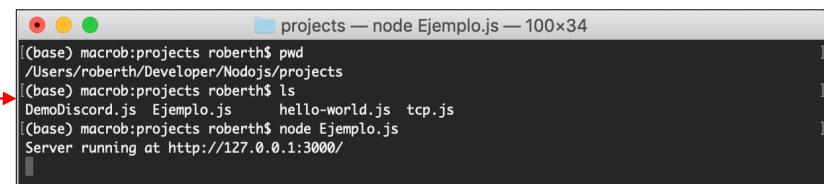
```
Nodojs — sudo — 100x34
npm help npm involved overview
Specify configs in the ini-formatted file:
  /Users/roberth/.npmrc
or on the command line via: npm <command> --key value
Config info can be viewed via: npm help config

npm@6.14.13 /usr/local/lib/node_modules/npm
(base) macrob:Nodojs roberth$ npm install -g n
npm [WARN] checkPermissions Missing write access to /usr/local/lib/node_modules/n
npm [WARN] checkPermissions Missing write access to /usr/local/lib/node_modules/n Version to install
npm [ERR] code EACCES
npm [ERR] syscall access
npm [ERR] path /usr/local/lib/node_modules/n
npm [ERR] errno -13
npm [ERR] Error: EACCES: permission denied, access '/usr/local/lib/node_modules/n'
npm [ERR] [Error: EACCES: permission denied, access '/usr/local/lib/node_modules/n'] {
npm [ERR]   [Error: EACCES: permission denied, access '/usr/local/lib/node_modules/n'] {
npm [ERR]     errno: -13,
npm [ERR]     code: 'EACCES',
npm [ERR]     syscall: 'access',
npm [ERR]     path: '/usr/local/lib/node_modules/n'
npm [ERR]   }
npm [ERR] }
npm [ERR] The operation was rejected by your operating system.
npm [ERR] It is likely you do not have the permissions to access this file as the current user
npm [ERR] If you believe this might be a permissions issue, please double-check the
npm [ERR] permissions of the file and its containing directories, or try running
npm [ERR] the command again as root/Administrator.

npm [ERR] A complete log of this run can be found in:
npm [ERR]   /Users/roberth/.npm/_logs/2021-07-19T17_00_38_854Z-debug.log
(base) macrob:Nodojs roberth$ sudo npm install -g n
Password: ?
```

Escribiendo un ejercicio en Node → Ejemplo.js

- mkdir Miproyecto
- Crear Ejemplo.js
- Ejecutar el comando
→ node Ejemplo.js
- Ir al navegador y abrir la URL
→ http://127.0.0.1:3000/ o http://localhost:3000/

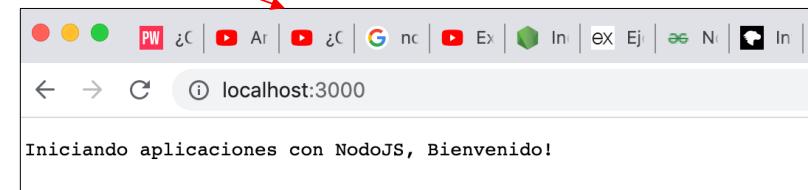


```
(base) macrob:projects roberth$ pwd
/Users/roberth/Developer/Nodojs/projects
(base) macrob:projects roberth$ ls
DemoDiscord.js Ejemplo.js hello-world.js tcp.js
(base) macrob:projects roberth$ node Ejemplo.js
Server running at http://127.0.0.1:3000/
```

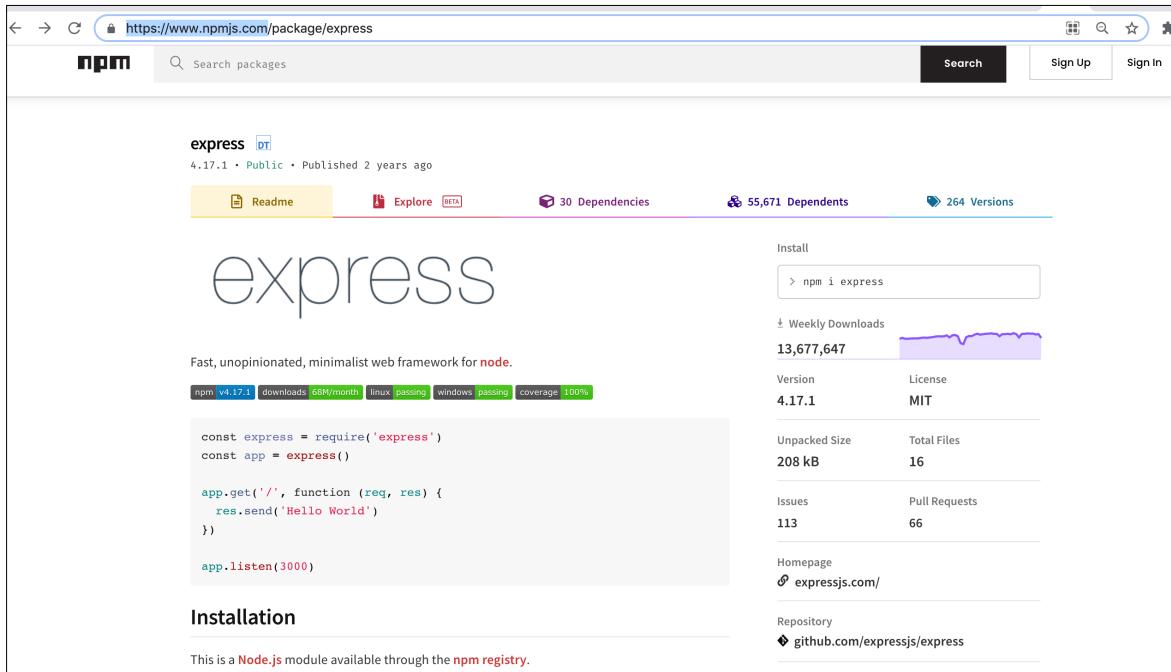
Ejemplo.js

```
const http = require('http');
const hostname = '127.0.0.1';
const port = 3000;
const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Iniciando aplicaciones con NodoJS, Bienvenido!\n');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```



Express.js o simplemente Express



The screenshot shows the npmjs.com package page for 'express'. The page header includes the npm logo and search bar. The main content area features the title 'express' in large font, followed by a brief description: 'Fast, unopinionated, minimalist web framework for node.' Below this are sections for 'Readme', 'Explore (BETA)', '30 Dependencies', '55,671 Dependents', and '264 Versions'. A code snippet example is provided:

```
const express = require('express')
const app = express()

app.get('/', function (req, res) {
  res.send('Hello World')
})

app.listen(3000)
```

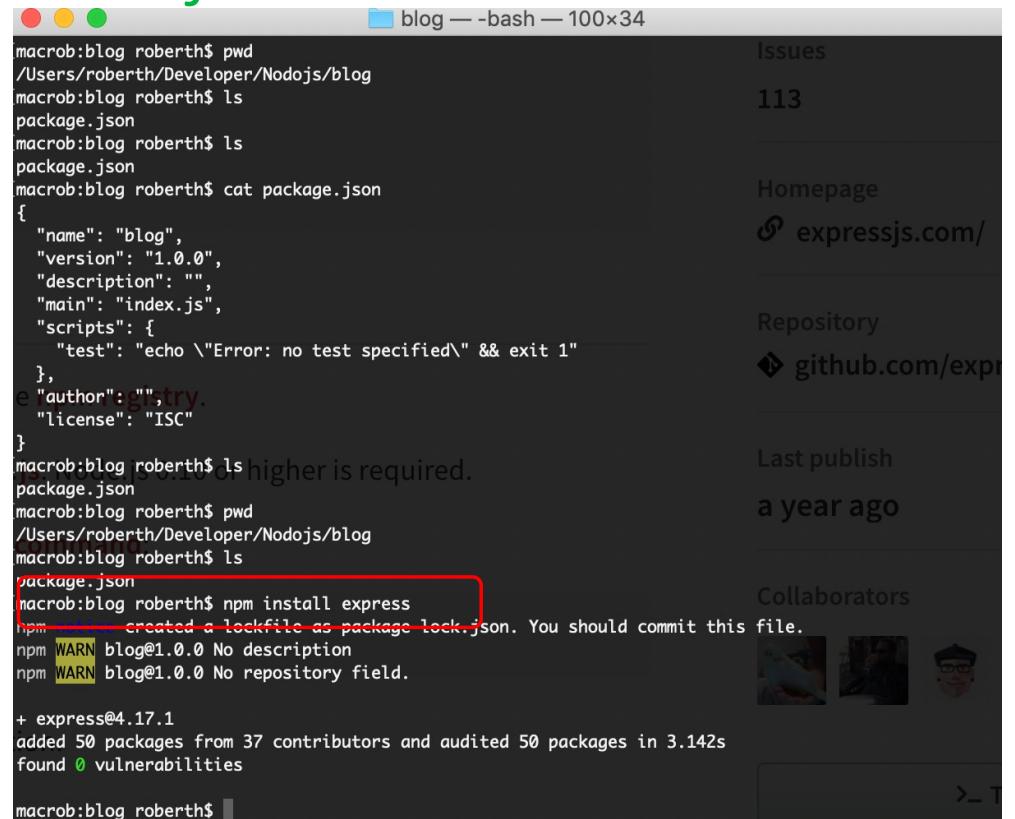
The 'Installation' section contains the command: `> npm i express`. To the right, there's a 'Weekly Downloads' chart showing 13,677,647, version information (4.17.1), and other metrics like Unpacked Size (208 kB), Total Files (16), Issues (113), Pull Requests (66), Homepage (expressjs.com/), and Repository (github.com/expressjs/express).

Express: Es un marco de trabajo o framework, que actua como una capa ligera sobre el servidor web de Node.js para facilitar el desarrollo de aplicaciones web.

Instalando el paquete Express.js

Ejecutar:

- **npm init** → crea un archivo package.json
→ ver el contenido dependencias **cat package.json**
- **npm install express** → instala express



```

macrob:blog roberth$ pwd
/Users/roberth/Developer/Nodojs/blog
macrob:blog roberth$ ls
package.json
macrob:blog roberth$ ls
package.json
macrob:blog roberth$ cat package.json
{
  "name": "blog",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \\"Warning: no test specified\\" && exit 1"
  },
  "author": "Roberth Gómez, try.",
  "license": "ISC"
}
macrob:blog roberth$ ls
macrob:blog roberth$ npm install express
npm WARN blog@1.0.0 No description
npm WARN blog@1.0.0 No repository field.

+ express@4.17.1
added 50 packages from 37 contributors and audited 50 packages in 3.142s
found 0 vulnerabilities

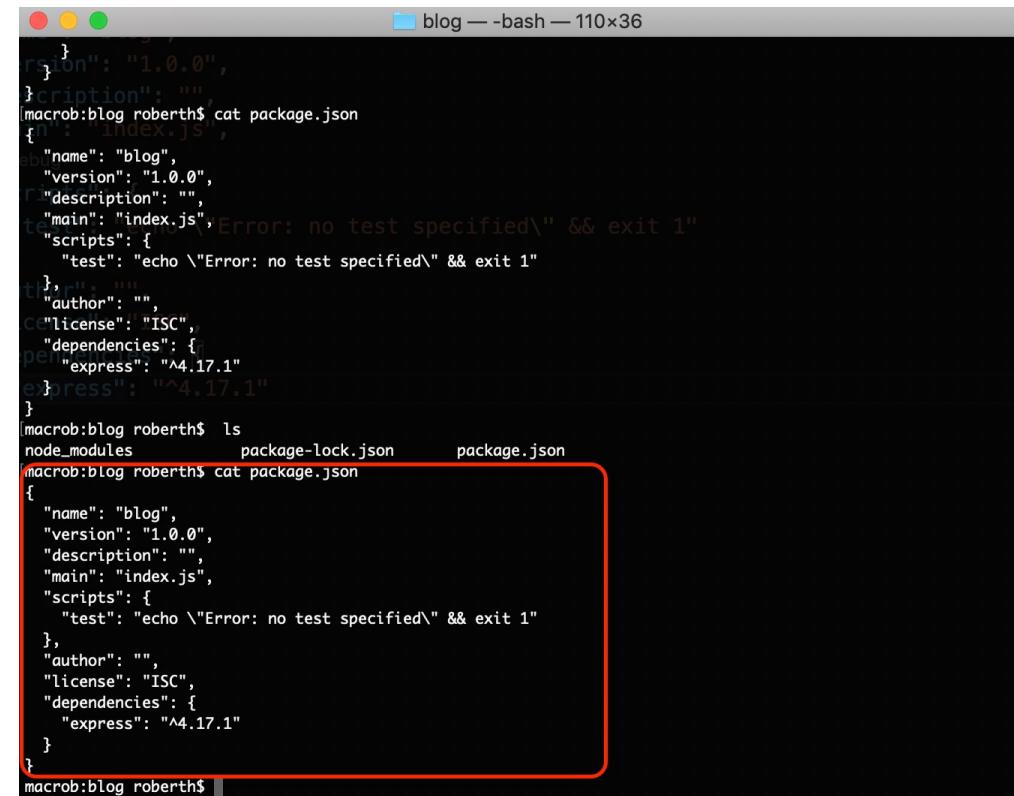
macrob:blog roberth$ 

```

The screenshot shows a terminal window titled "blog — bash — 100x34". The terminal displays the command "npm install express" being run, which installs the express package version 4.17.1. The output also includes several warning messages about the lack of a description and repository fields in the package.json file.

Ver dependencias instaladas

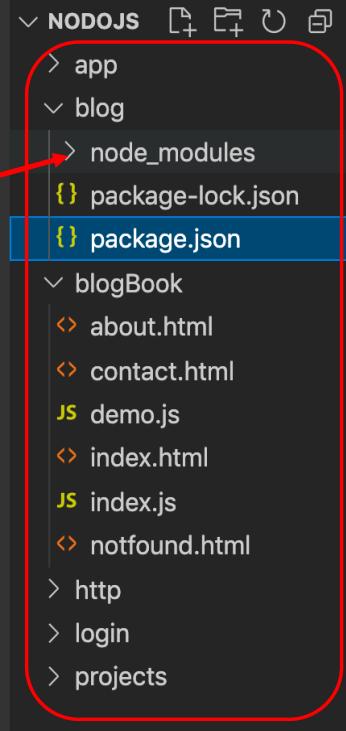
```
macrob:blog roberth$ cat package.json
{
  "name": "blog",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \\\"Error: no test specified\\\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "express": "^4.17.1"
  }
}
```



```
blog — bash — 110x36
[macrob:blog roberth$ cat package.json
{
  "name": "blog",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \\\"Error: no test specified\\\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "express": "4.17.1"
  }
}
[macrob:blog roberth$ ls
node_modules          package-lock.json      package.json
[macrob:blog roberth$ cat package.json
{
  "name": "blog",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \\\"Error: no test specified\\\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "express": "4.17.1"
  }
}
[macrob:blog roberth$ ]
```

Dependencias del aplicación → package-lock.json

Package Express



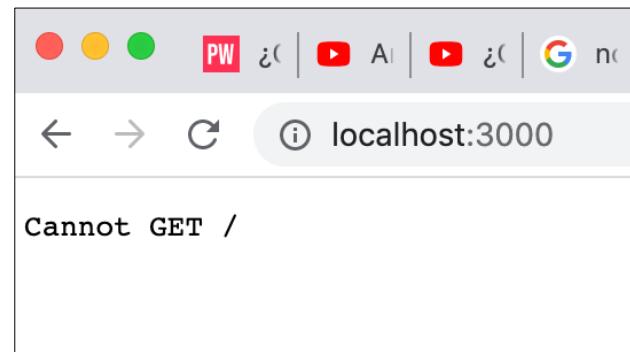
```
1  {
2    "name": "blog",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "test": "echo \\\"Error: no test specified\\\" && exit 1"
8    },
9    "author": "",
10   "license": "ISC",
11   "dependencies": {
12     "express": "^4.17.1"
13   }
14 }
```

Ejemplo con Express

```
const express = require('express');
const app = express();
app.listen(3000, () => {
  console.log("App escuchando en el puerto 3000");
});
```

Ejecutar → node Ejemplo.js

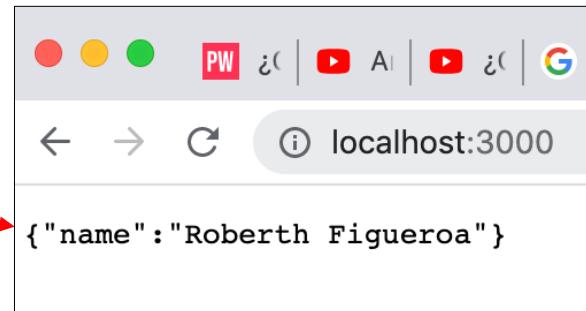
```
[macrob:blog roberth$ node index.js
App escuchando en el puerto 3000]
```



Ejemplo con Express

```
const express = require('express');
const app = express();
app.listen(3000, () => {
    console.log("App escuchando en el puerto 3000");
});
app.get('/',(req,res)=>{
    res.json({
        name:'Roberth Figueroa'
    });
});
```

Se retorna un JSON al cliente gracias a Express



Enrutado de llamadas (*called Routing*)

Se devuelve la respuesta especificada según la petición o request.

```

1  const http = require('http');
2  const fs = require('fs');
3
4  const HomePage = fs.readFileSync('index.html');
5  const aboutPage = fs.readFileSync('about.html');
6  const contactPage = fs.readFileSync('contact.html');
7  const notFoundPage = fs.readFileSync('notfound.html');
8
9  const server = http.createServer((req, res) => {
10    if (req.url === '/about') {
11      res.end(aboutPage);
12    } else if (req.url === '/contact') {
13      res.end(contactPage);
14    } else if (req.url === '/') {
15      res.end(homePage);
16    } else {
17      res.writeHead(404);
18      res.end(notFoundPage);
19    }
20  });
21  server.listen(3000);

```



Sin Express

```

1  const express = require('express');
2  const path = require('path');
3
4  const app = express();
5  app.listen(3000, () => {
6    console.log("App escuchando en el puerto 3000");
7  });
8  app.get('/', (req, res) => {
9    res.sendFile(path.resolve(__dirname, 'index.html'));
10 };
11 app.get('/contact', (req, res) => {
12   res.sendFile(path.resolve(__dirname, 'contact.html'));
13 };
14 app.get('/about', (req, res) => {
15   res.sendFile(path.resolve(__dirname, 'about.html'));
16 };
17 app.get('/notfound', (req, res) =>{
18   res.sendFile(path.resolve(__dirname, 'notfound.html'));
19 });

```

Con Express

```

const express = require('express');
const path = require('path');
const app = express();
app.listen(3000, () => {
  console.log("App escuchando en el puerto 3000");
  app.get('/', (req, res) => {
    res.sendFile(path.resolve(__dirname, 'index.html'));
  });
  app.get('/contact', (req, res) => {
    res.sendFile(path.resolve(__dirname, 'contact.html'));
  });
  app.get('/about', (req, res) => {
    res.sendFile(path.resolve(__dirname, 'about.html'));
  });
  app.get('/notfound', (req, res) =>{
    res.sendFile(path.resolve(__dirname, 'notfound.html'));
  });
});

```



Contacto



Universidad
Nacional
de Loja



Ing. Roberth Figueroa Díaz M.Sc.



Carrera Computación



roberth.figueroa@unl.edu.ec



www.unl.edu.ec



unl

Universidad
Nacional
de Loja

Gracias por su atención

REDES **unl** OFICIALES



Universidad Nacional De Loja - UNL



@UnlOficial



UnlOficial



UnlOficial

Educamos para transformar