
 GOBIERNO DEL ESTADO DE MÉXICO	MANUAL DE PRÁCTICAS FO-TESJI-11100-12		 TESJI TECNOLÓGICO DE ESTUDIOS SUPERIORES JILOTEPEC
NOMBRE DE LA PRÁCTICA:	Practica Transferencia De Archivos		No. 1
ASIGNATURA:	Fundamentos de telecomunicaciones	CARRERA: ISIC	Unidad: III
ALUMNO:	Vanesa Hernández Martínez y Jesús Navarrete Martínez		

1- Competencias Específicas:

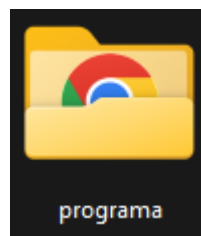
2- Desarrollo con: HTML, CSS, Javascript, NODE.JS y cable ethernet

3- Desarrollo de la Practica:

1. Instalamos la libreria NODE.JS para poder crear un servidor que escuche los puertos ethernet.



2. Creamos una carpeta donde guardaremos el archivo del servidor.



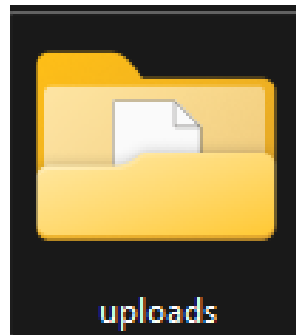
3. Dentro de esa carpeta abrimos CMD y ejecutamos el siguiente comando: `npm install express express-fileupload`

```
Windows PowerShell
Copyright (c) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PSWindows

PS C:\Users\Vanessa Hernandez\Desktop\programa> npm install express express-fileupload
```

4. Creamos una carpeta dentro de la carpeta anteriormente creada en la cual se almacenarán los archivos recibidos en la transferencia al utilizar la aplicación.



5. Dentro de la carpeta creamos un archivo javascript para la ejecución del servidor.

```
JS server.js  X  <> index.html
JS server.js > ...
1  const express = require('express');
2  const fileUpload = require('express-fileupload');
3  const app = express();
4  const PORT = 3000;
5
6  app.use(fileUpload());
7
8  app.post('/upload', (req, res) => {
9    const uploadedFile = req.files.file;
10    uploadedFile.mv('uploads/' + uploadedFile.name, (err) => {
11      if (err) {
12        return res.status(500).send(err);
13      }
14      res.send('Archivo subido correctamente!');
15    });
16  });
17
18  app.listen(PORT, () => {
19    console.log('Servidor escuchando en el puerto ${PORT}');
20  });
```

6. Creamos la interfaz de nuestra aplicación con ayuda de html y css, la cual también contiene una sección desarrollada con javascript para validar datos y transferir archivos.


```
JS server.js  index.html ●
index.html > ...
1  <!doctype html>
2  <html lang="es">
3  <head>
4    <title>FIBUNACCI</title>
5    <meta charset="utf-8">
6    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
7    <link rel="stylesheet" href="https://unicons.iconscout.com/release/v2.1.9/css/unicons.css">
8    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.5.0/css/bootstrap.min.css">
9    <link rel="stylesheet" href="estilo.css">
10 </head>
11 <body>
12
13
14 <div id="stars"></div>
15 <div id="stars2"></div>
16 <div id="stars3"></div>
17 <div class="section">
18   <div class="container">
19     <div class="row full-height justify-content-center">
20       <div class="col-12 text-center align-self-center py-5">
21         <div class="section pb-5 pt-5 pt-sm-2 text-center">
22           <h6 class="mb-0 pb-3"><span>Ip</span><span>Subir</span></h6>
23           <input class="checkbox" type="checkbox" id="reg-log" name="reg-log"/>
24           <label for="reg-log"></label>
25           <div class="card-3d-wrap mx-auto">
26             <div class="card-3d-wrapper">
27               <div class="card-front">
28                 <div class="center-wrap">
29                   <div class="section text-center">
30
31
32                   <h4 class="mb-4 pb-3">:::Bienvenido a FAST TRANSFER::</h4>
33                   <H6>Desarrollado por VANSUS</H6>
34
35                   <div class="form-group">
36
37                     <input id="ipInput" class="control" type="text" placeholder="Ingrese la direccion ip del ser
38                     <i class="input-icon uil uil-pen"></i>
39
40
41                   </div>
42
43                   </div>
44
45                   </div>
46
47                   </div>
48                   </div>
49                   <div class="card-back">
50                     <div class="center-wrap">
51                       <div class="section text-center">
52                         <h4 class="mb-3 pb-3">Comparte tus archivos de forma rapida y segura:</h4>
53                         <h5 id="resultado"></h5>
54                         <input type="file" id="fileInput" class="control"/>
55                         <button class="control" onclick="uploadFile()">Subir Archivo</button>
56
```

```

61 |         </div>
62 |     </div>
63 | </div>
64 | </div>
65 | </div>
66 |
67 | </div>
68 | </div>
69 | </div>
70 | </div>
71 | </div>
72 | </div>
73 |
74 | <script>
75 | function uploadFile() {
76 |     var fileInput = document.getElementById('fileInput');
77 |     var file = fileInput.files[0];
78 |     var ipInput = document.getElementById('ipInput');
79 |     var ipAddress = ipInput.value; // Obtener la dirección IP ingresada por el usuario
80 |
81 |     // Verificar si se seleccionó un archivo
82 |     if (!file) {
83 |         alert('Por favor, selecciona un archivo');
84 |         return;
85 |     }
86 |
87 |     // Verificar si se ingresó una dirección IP
88 |     if (ipAddress.trim() === '') {
89 |         alert('Por favor, ingresa una dirección IP');
90 |         return;
91 |     }
92 |
93 |     var formData = new FormData();
94 |     formData.append('file', file);
95 |
96 |     // Realizar la solicitud POST al servidor con la dirección IP proporcionada
97 |     fetch('http://' + ipAddress + ':3000/upload', {
98 |         method: 'POST',
99 |         body: formData
100 |     })
101 |     .then(response => response.text())
102 |     .then(data => alert(data))
103 |     .catch(error => console.error('Error:', error));
104 | }
105 | </script>
106 | </body>
107 | </html>

```

- Activamos el servidor que hemos creado, abriendo la terminal desde la carpeta en donde se encuentra el `server.js` y escribimos el siguiente comando.



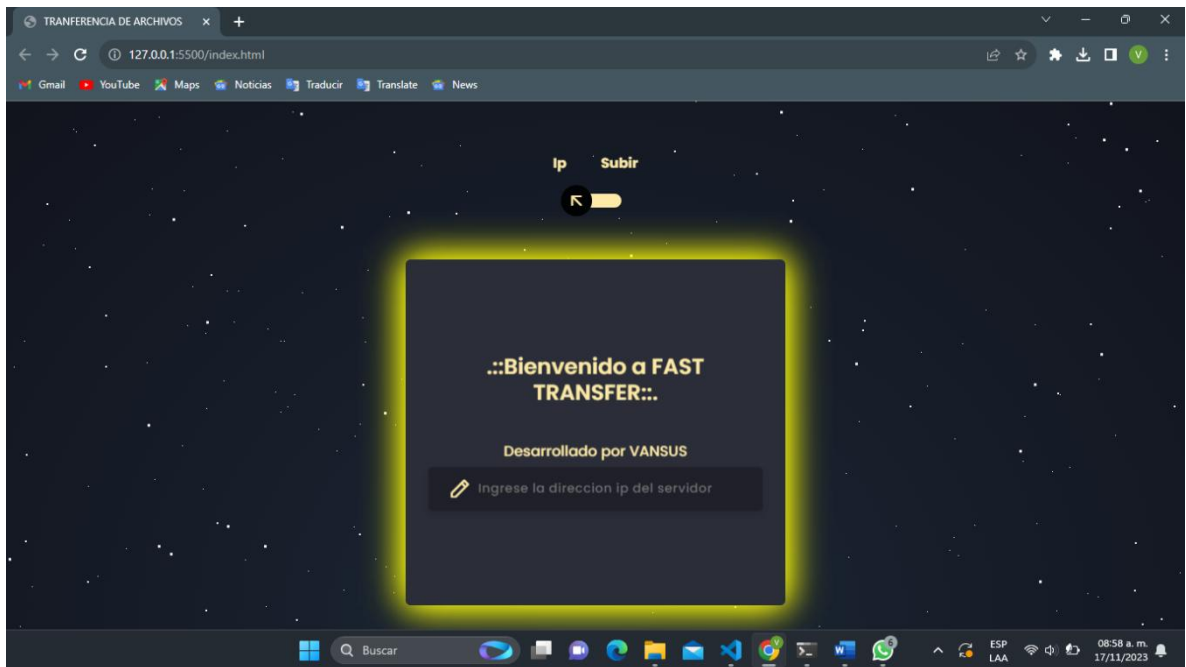
The screenshot shows a Windows PowerShell terminal window. The title bar reads "Windows PowerShell". The window content displays the following text:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PSWindows

PS C:\Users\Vanesa Hernandez\Desktop\programa> node server.js
Servidor escuchando en el puerto ${PORT}
```

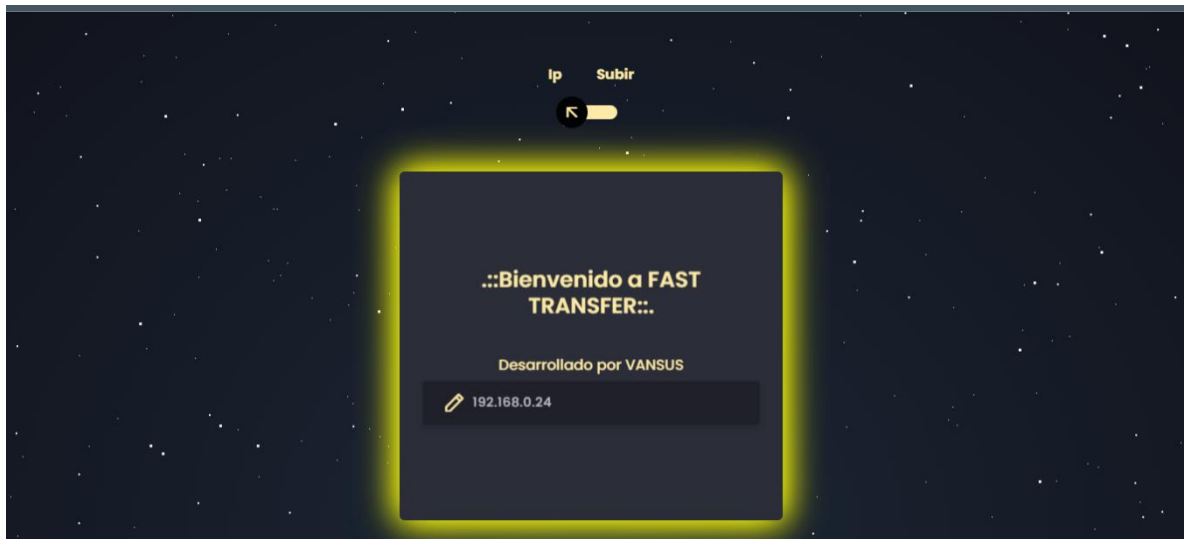
8. Una vez que el servidor fue creado correctamente abrimos la interfaz de nuestra aplicación en un navegador.



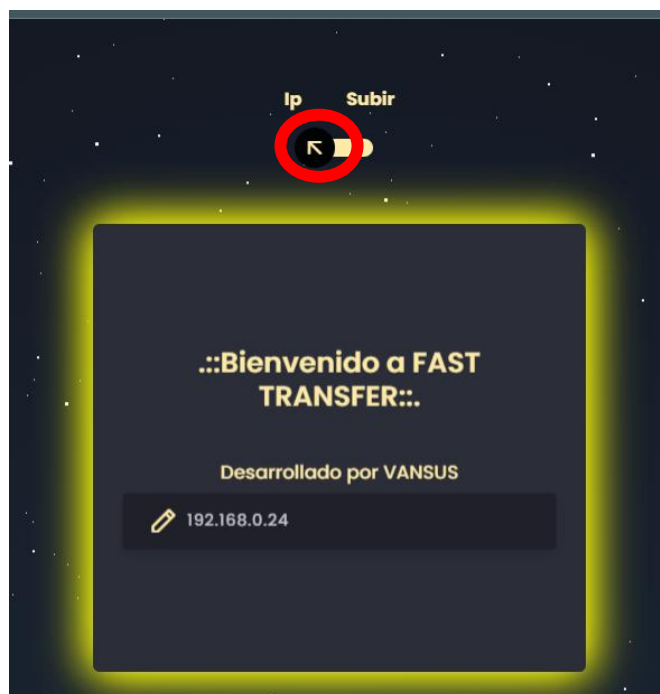
NOTA: Es importante realizar estos pasos en ambas computadoras en las que se desea enviar archivos

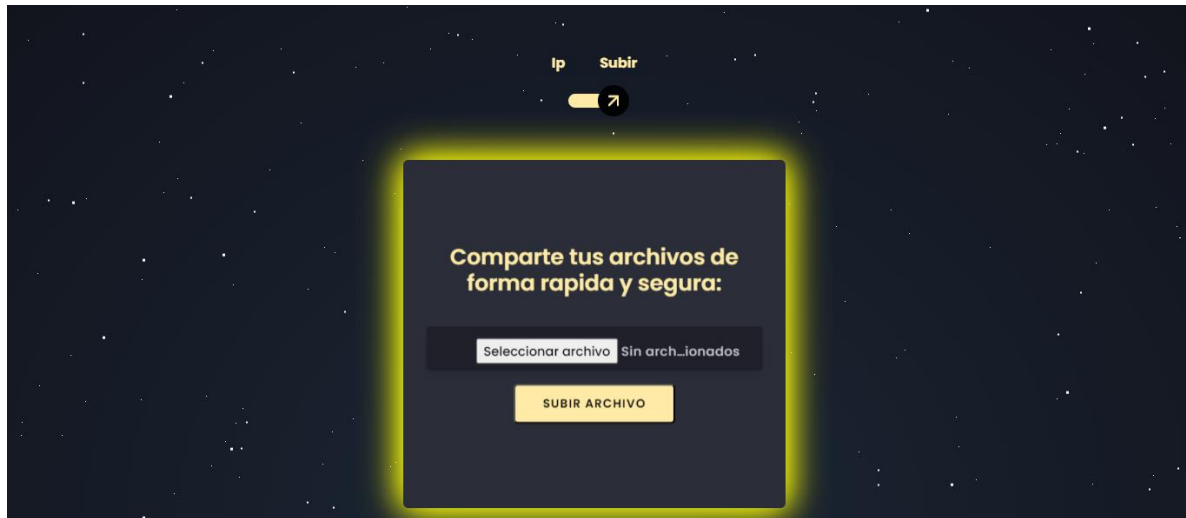
MANUAL DE USUARIO

1. Primero conectamos el cable ethernet de computadora a computadora.
2. Una vez dentro de la interfaz, encontraremos una sección para ingresar la ip de la computadora a quien deseamos enviar archivos.

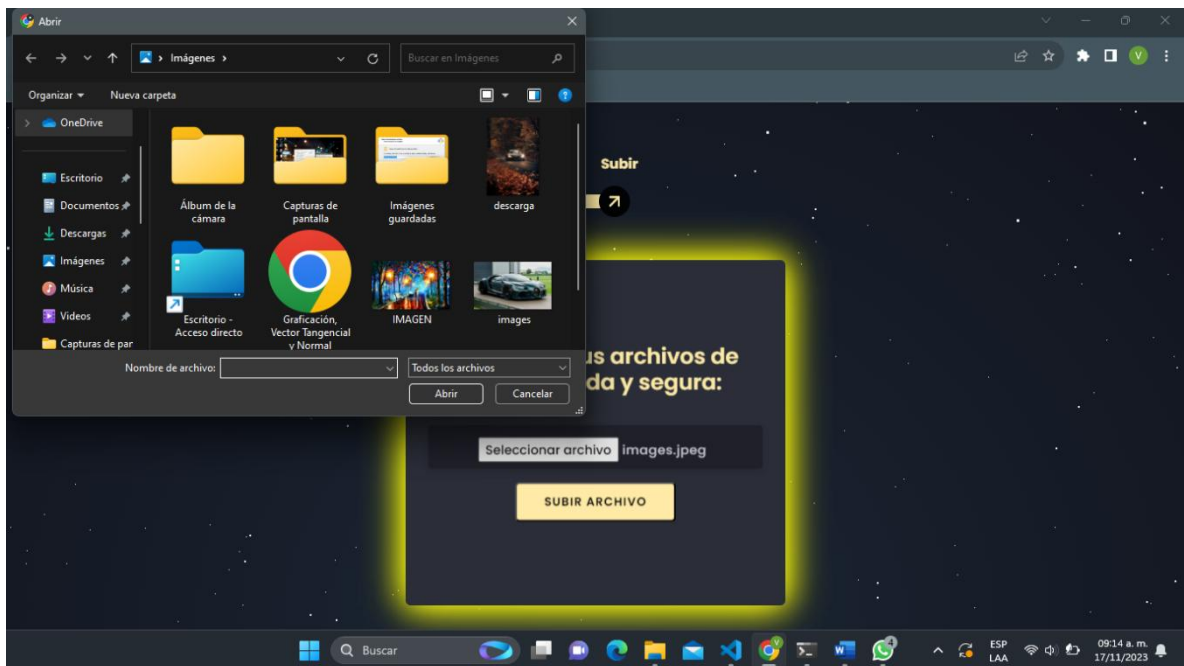


3. Para continuar con el proceso de transferencia de archivos, damos clic en la flecha de la parte superior.

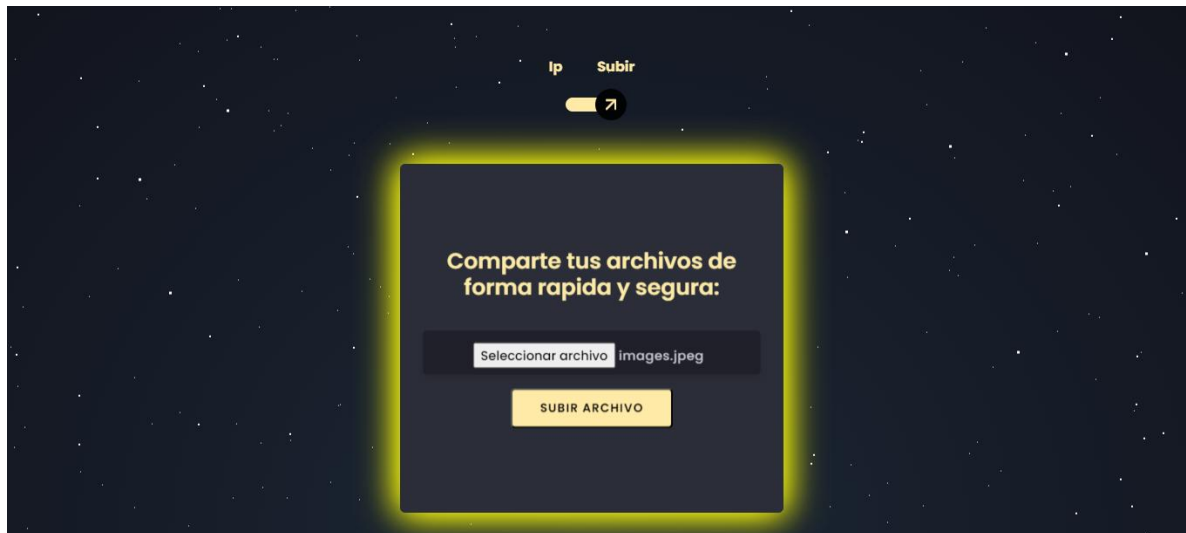




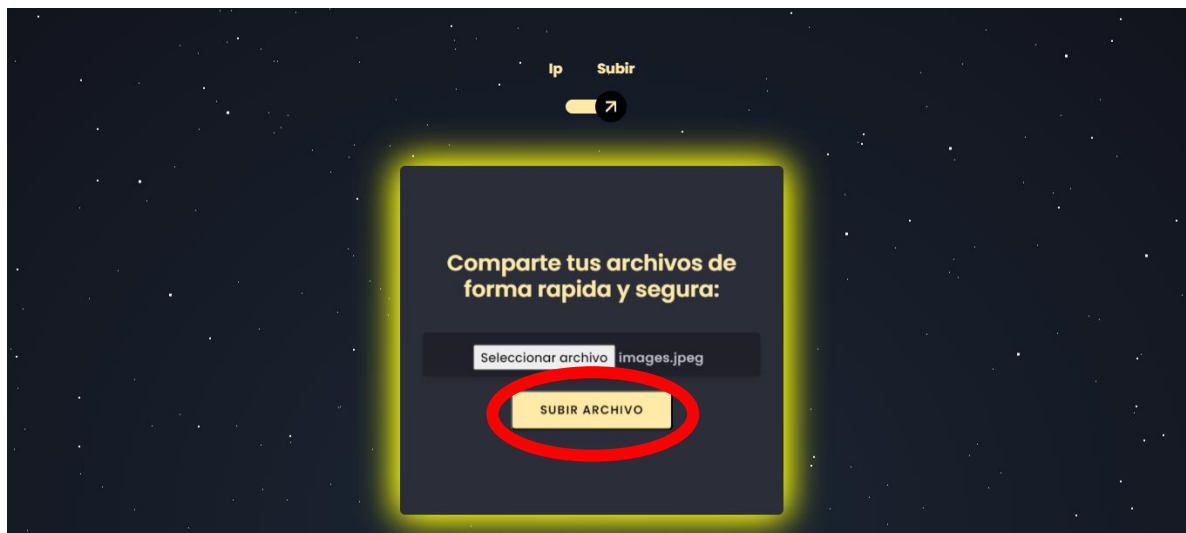
4. En el siguiente apartado es donde se subirá el archivo a transferir, para ello, damos clic en seleccionar archivo, lo cual nos abrirá el explorador para seleccionar el archivo que deseamos enviar.



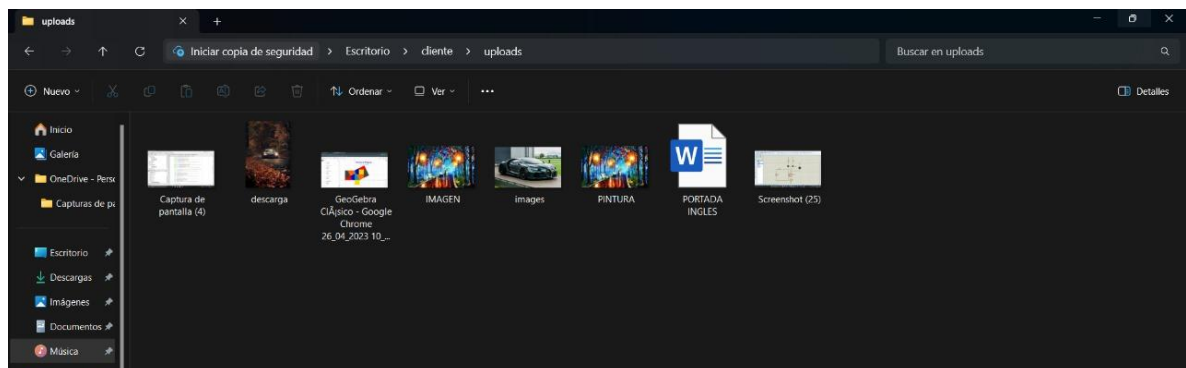
5. Una vez cargado se visualizará el nombre del archivo y esto nos indicará que ya está listo para ser transferido.



6. Damos clic en subir archivo



7. Una vez realizados todos los pasos anteriores, nos dirigimos a la carpeta llamada **uploads** del receptor, en la cual se visualizará el archivo enviado.



CONCLUSION

La transferencia de archivos es una acción muy común dentro de nuestro mundo, continuamente tenemos que compartir archivos entre amigos, familiares o compañeros de trabajo y escuela, este proceso se puede realizar desde una computadora a otra utilizando un cable ethernet, pero es un proceso complejo por lo que nuestra finalidad es facilitarle a los usuarios la transferencia de archivos de manera que no se necesite tener mucha experiencia en el medio para poder realizar esta acción tan básica pero tan importante a la vez en nuestro mundo actual.