

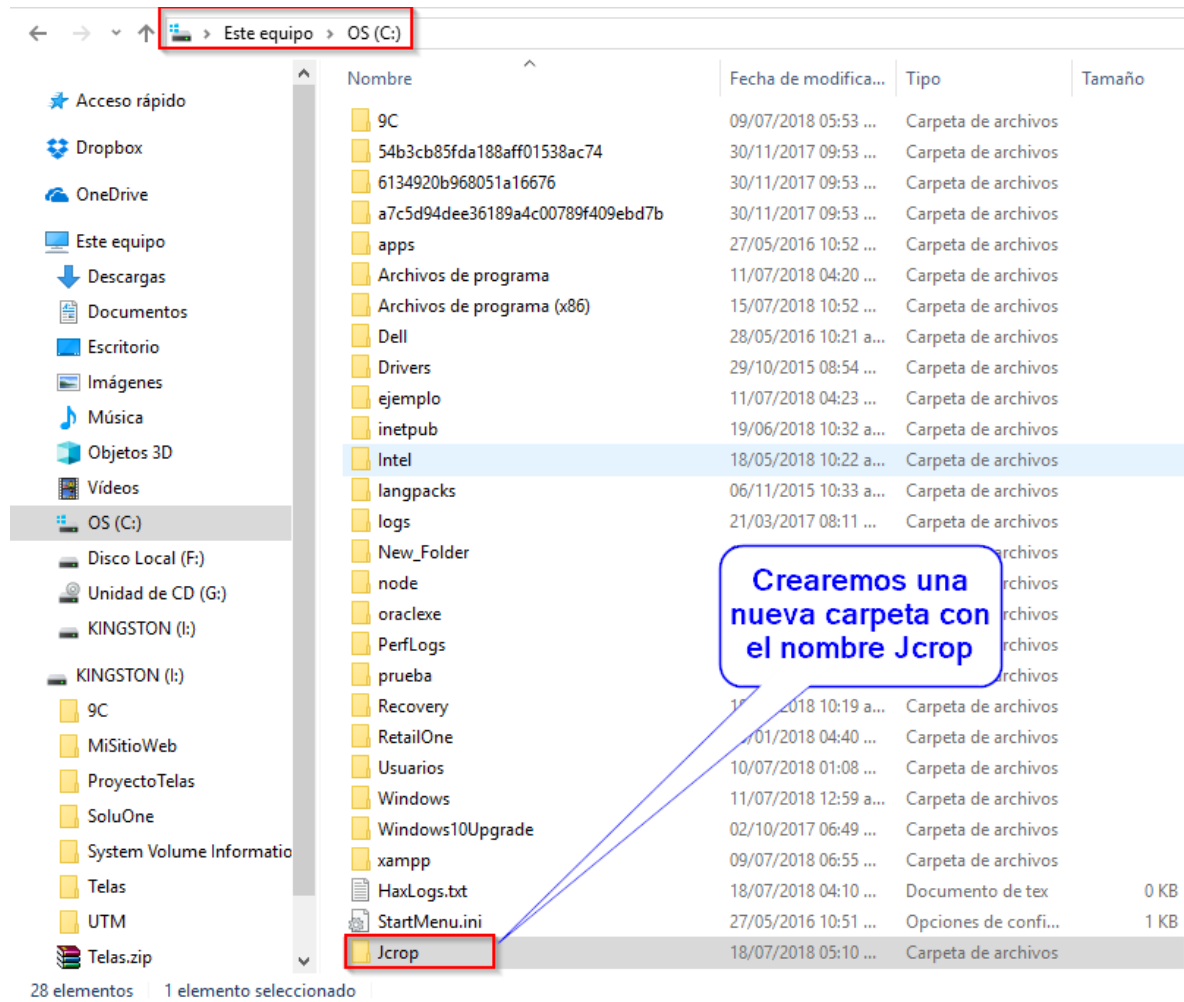


Walter Escamilla Pech

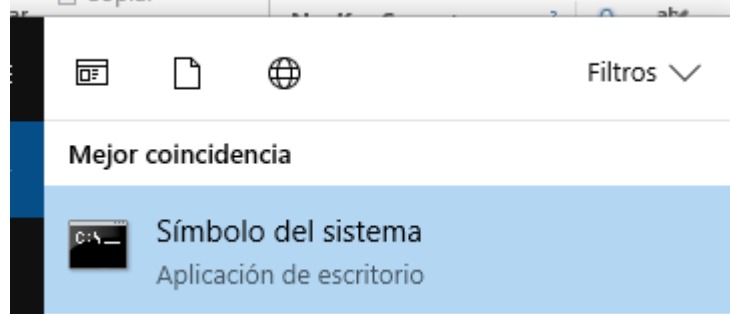


**Manual para la creación de un Jcrop de imágenes
usando las tecnologías Node.js y Express**

1.- Primero nos dirigiremos al disco local c y crearemos una nueva carpeta con el nombre Jcrop en donde se creará el proyecto



2.- Ahora ejecutaremos el programa Símbolo del sistema



3.- Escribiremos el siguiente comando:

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.17134.165]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Walter>cd C:\Jcrop

C:\Jcrop>
```

Escribimos la ruta del proyecto

4.- Ahora instalaremos express

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.17134.165]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Walter>cd C:\Jcrop

C:\Jcrop>npm install express
npm WARN saveError ENOENT: no such file or directory, open 'C:\Jcrop\package.json'
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN enoent ENOENT: no such file or directory, open 'C:\Jcrop\package.json'
npm WARN Jcrop No description
npm WARN Jcrop No repository field.
npm WARN Jcrop No README data
npm WARN Jcrop No license field.

+ express@4.16.3
added 50 packages from 47 contributors and audited 119 packages in 95.01s
found 0 vulnerabilities

C:\Jcrop>
```

5.- Usamos la siguiente instrucción, para crear un nuevo proyecto

```
C:\Jcrop>npm install express-generator -g
C:\Users\Walter\AppData\Roaming\npm\express -> C:\Users\Walter\AppData\Roaming\npm\node_modules\express-generator\bin\express-cli.js
+ express-generator@4.16.0
updated 1 package in 2.089s
```

6.- Ahora crearemos la estructura de nuestra aplicación web usando el siguiente comando

```
C:\Jcrop>express --view-ejs
```

```

create : public\
create : public\javascripts\
create : public\images\
create : public\stylesheets\
create : public\stylesheets\style.css
create : routes\
create : routes\index.js
create : routes\users.js
create : views\
create : views\error.ejs
create : views\index.ejs
create : app.js
create : package.json
create : bin\
create : bin\www

install dependencies:
  > npm install

run the app:
  > SET DEBUG=calculadoraimc:* & npm start

```

7.- Ahora instalaremos la dependencia cookie-parser para poder funcionar

```

C:\Jcrop>npm install cookie-parser
+ cookie-parser@1.4.3
added 1 package from 2 contributors in 1.444s

```

8.- Ahora instalaremos “Morgan” con el siguiente comando:

```

C:\Jcrop>npm install morgan
+ morgan@1.9.0
added 3 packages from 2 contributors in 1.654s

```

9.- Por ultimo instalaremos el motor de plantillas ejs para trabajar con las views

```

C:\Jcrop>npm install ejs
+ ejs@2.6.1
added 1 package from 2 contributors in 1.644s

```

10.- Ahora escribimos el siguiente comando para correr nuestra aplicación web usando “npm start” y verificamos que no marque error

```

C:\Jcrop>npm start
> sitioweb@0.0.0 start C:\9C\sitioweb
> node ./bin/www

```

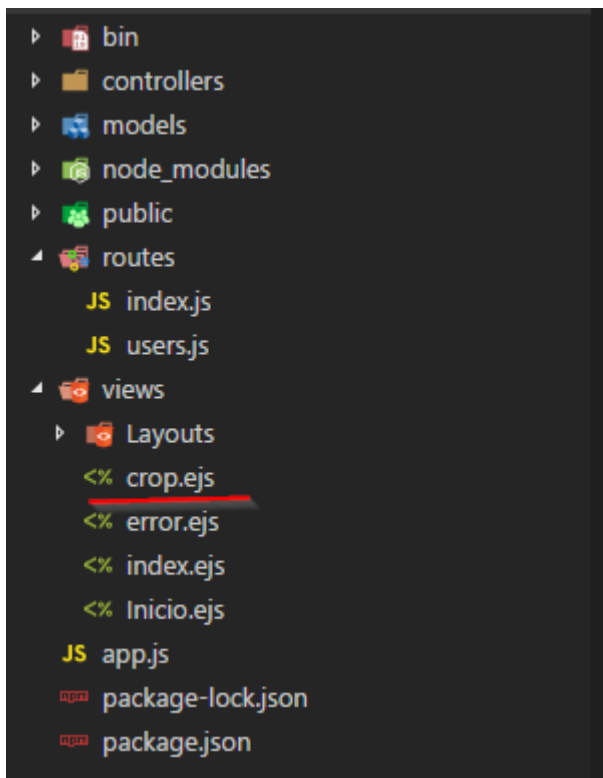
11.- Para acceder a su aplicación escriba en el navegador la siguiente URL “local host: 3000” si en el navegador visualiza el siguiente contenido significa que ya está listo para poder trabajar con Express.js



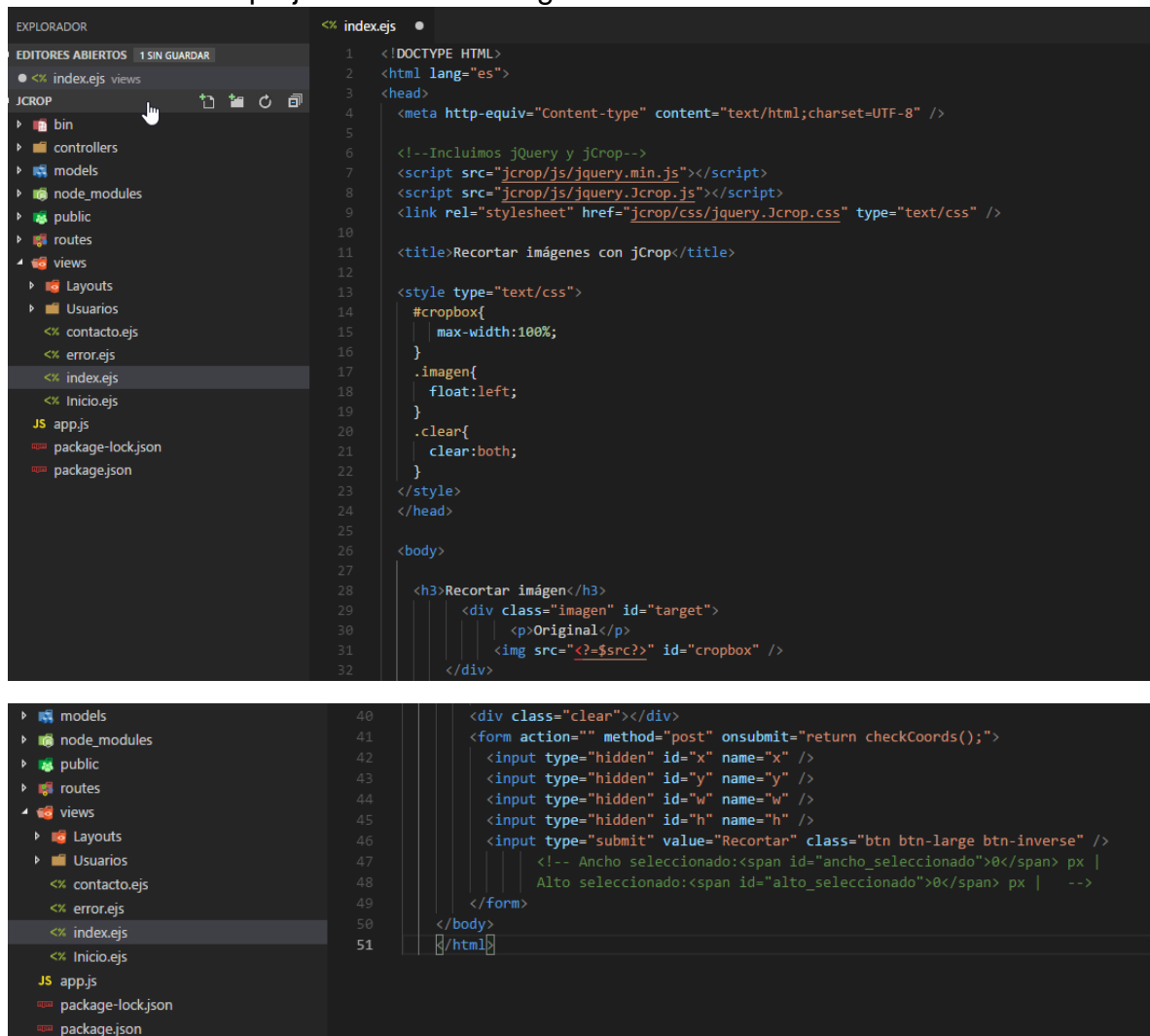
Express

Welcome to Express

12.- En la carpeta “views” crearemos la vista para crop llamada y “crop.ejs”, que será la encargada de la lógica del crop:

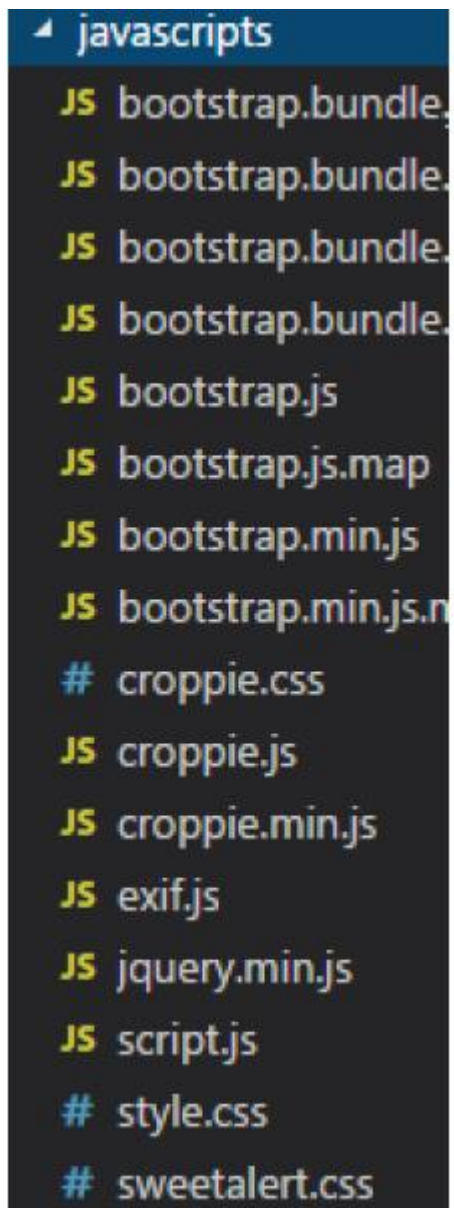


13.- La clase "crop.ejs" contendrá lo siguiente:

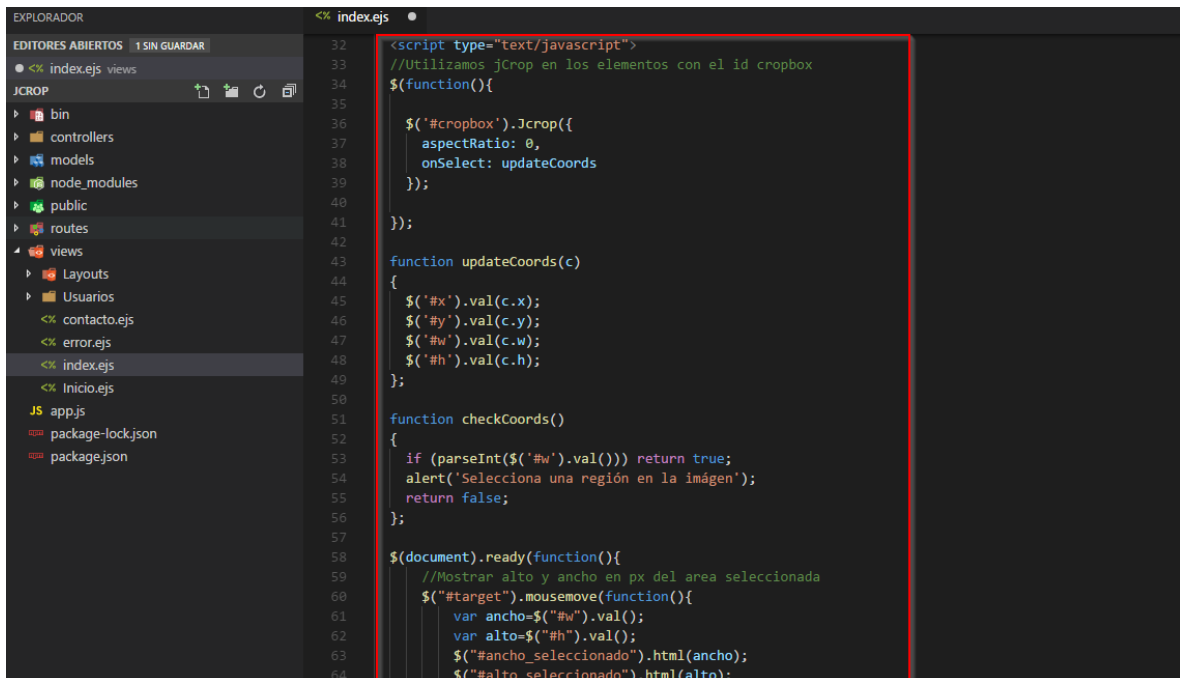


```
1 <!DOCTYPE HTML>
2 <html lang="es">
3 <head>
4   <meta http-equiv="Content-type" content="text/html; charset=UTF-8" />
5
6   <!-- Incluimos jQuery y jCrop -->
7   <script src="jcrop/js/jquery.min.js"></script>
8   <script src="jcrop/js/jquery.Jcrop.js"></script>
9   <link rel="stylesheet" href="jcrop/css/jquery.Jcrop.css" type="text/css" />
10
11   <title>Recortar imágenes con jCrop</title>
12
13   <style type="text/css">
14     #cropbox{
15       max-width:100%;
16     }
17     .imagen{
18       float:left;
19     }
20     .clear{
21       clear:both;
22     }
23   </style>
24 </head>
25
26 <body>
27
28   <h3>Recortar imagen</h3>
29   <div class="imagen" id="target">
30     <p>Original</p>
31     
32   </div>
33
34   <div class="clear"></div>
35   <form action="" method="post" onsubmit="return checkCoords();">
36     <input type="hidden" id="x" name="x" />
37     <input type="hidden" id="y" name="y" />
38     <input type="hidden" id="w" name="w" />
39     <input type="hidden" id="h" name="h" />
40     <input type="submit" value="Recortar" class="btn btn-large btn-inverse" />
41     <!-- Ancho seleccionado:<span id="ancho_seleccionado">0</span> px |
42          Alto seleccionado:<span id="alto_seleccionado">0</span> px | -->
43   </form>
44
45 </body>
46 </html>
```

14.- Agregaremos las librerías que podemos encontrar aquí <https://foliotek.github.io/Croppie/>. O ejecutando npm install croppie



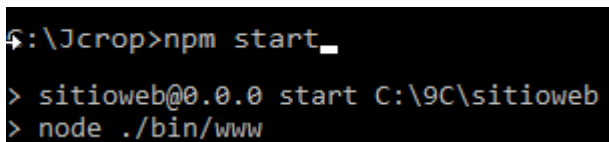
15.- Ahora crearemos el script el cual será el encargado de hacer los recortes de la imagen



The screenshot shows the VS Code editor with the Explorer sidebar on the left. The Explorer sidebar shows the project structure with folders like bin, controllers, models, node_modules, public, routes, and views. The views folder is expanded, showing files like layouts, Usuarios, contacto.ejs, error.ejs, index.ejs, and Inicio.ejs. The index.ejs file is selected and its content is displayed in the editor. The code is a JavaScript script that initializes jCrop and defines functions for updating and checking coordinates.

```
32 <script type="text/javascript">
33 //Utilizamos jCrop en los elementos con el id cropbox
34 $(function(){
35
36     $('#cropbox').Jcrop({
37         aspectRatio: 0,
38         onSelect: updateCoords
39     });
40
41 });
42
43 function updateCoords(c)
44 {
45     $('#x').val(c.x);
46     $('#y').val(c.y);
47     $('#w').val(c.w);
48     $('#h').val(c.h);
49 };
50
51 function checkCoords()
52 {
53     if (parseInt($('#w').val())) return true;
54     alert('Selecciona una región en la imagen');
55     return false;
56 };
57
58 $(document).ready(function(){
59     //Mostrar alto y ancho en px del area seleccionada
60     $('#target').mousemove(function(){
61         var ancho=$('#w').val();
62         var alto=$('#h').val();
63         $('#ancho_seleccionado').html(ancho);
64         $('#alto_seleccionado').html(alto);
```

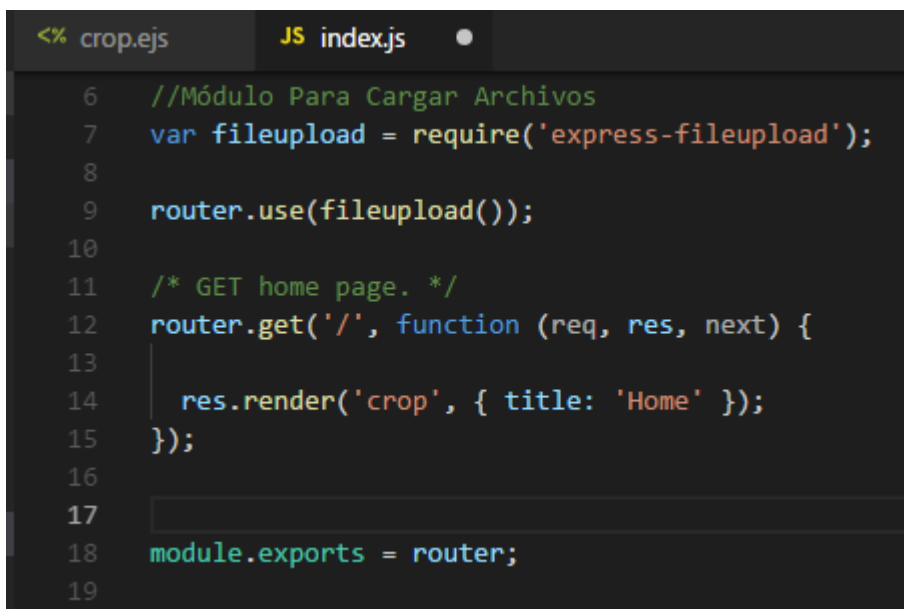
16.- Corremos el siguiente comando



The screenshot shows a terminal window with the following commands and output:

```
> \Jcrop> npm start
> sitioweb@0.0.0 start C:\9C\sitioweb
> node ./bin/www
```

17.- En la clase “index.js” de la carpeta routes agregaremos la siguiente línea de código para mostrar la vista que acabamos de crear



The screenshot shows the VS Code editor with the Explorer sidebar on the left. The Explorer sidebar shows the project structure with folders like bin, controllers, models, node_modules, public, routes, and views. The routes folder is expanded, showing files like index.js. The index.js file is selected and its content is displayed in the editor. The code is a JavaScript file that sets up the Express.js router and defines the home page route.

```
6 //Módulo Para Cargar Archivos
7 var fileupload = require('express-fileupload');
8
9 router.use(fileupload());
10
11 /* GET home page. */
12 router.get('/', function (req, res, next) {
13
14     res.render('crop', { title: 'Home' });
15 });
16
17
18 module.exports = router;
19
```


18.- Por ultimo nos dirigimos a nuestro explorador y tecleamos en el buscador <http://localhost:3000/>





localhost:3000



Seleccionamos el área a recortar

Recortar

Recortar imagen

Original



Recortada

