

Programas a usar:

- mysql workbench
- visual Studio code
- nodemon
- instalar nodejs (<https://nodejs.org/es/download/>)
- instalar angular:
 - node -v
 - npm -v
 - npm install -g typescript
 - tsc -v
 - npm install -g @angular/cli
 - ng version

1. Crear una base de datos en mysqlWorkbench1 con las siguientes lineas

```
create database if not exists db_basico;  
use db_basico;  
create table tb_equipo(  
id_equipo int not null auto_increment primary key,  
nombre varchar(50),  
logo varchar (200)  
);
```

2. Abrir Visual studio code y crear un nuevo proyecto nombre "APINODE"
3. Abrir la terminal de Visual Studio Code y ejecutamos el siguiente comando para crear un archivo json con la estructura del proyecto
"npm init -y"

y agregamos una nueva variable llamada "dev" y hará referencia a la api



```
{  
  "name": "apinode",  
  "version": "1.0.0",  
  "description": "",  
  "main": "index.js",  
  "scripts": {  
    "dev": "nodemon/index.js",  
    "test": "echo \\\"Error: no test specified\\\" && exit 1"  
  },  
  "keywords": [],  
  "author": "",  
  "license": "ISC",  
  "devDependencies": {  
    "nodemon": "^2.0.19"  
  },  
  "dependencies": {  
    "express": "^4.18.1",  
    "mysql": "^2.18.1"  
  }  
}
```

4. Creamos un nuevo archivo dentro del mismo proyecto con el nombre “index.js”
En este archivo se podrá iniciar el proyecto, contendrá el siguiente código:

```
require('./config/conexion');  
  
const express= require('express');  
const rutas = require('./rutas');  
const port =(process.env.port || 3000);  
  
const app= express();  
//admitir ripos dato  
app.use(express.json());  
  
app.set('port',port);  
  
//rutas  
app.use('/api',require('./rutas'))  
  
//iniciar xpress  
app.listen(app.get('port'),(error)=>{  
  if(error){  
    console.log('error al iniciar servidor '+error);  
  }else{  
    console.log('Servidor iniciado en el puerto: '+port);  
  }  
})
```

5. Creamos una nueva carpeta “config” y dentro de ésta un archivo para configurar la conexión a la base de datos “conexion.js” .

```
const mysql= require('mysql');  
const conexion = mysql.createConnection({  
  host: 'localhost',  
  user: 'root',  
  password: '1234',  
  port: '3306',  
  database: 'db_basico'  
});  
  
conexion.connect((err)=>{  
  if(err){  
    console.log('Hay un error: '+err)  
  }else{  
    console.log('Base de datos conectada')  
  }  
});  
  
module.exports=conexion;
```

6. Creamos un archivo de configuración de rutas, donde se irá creando los servicios, post, get, put y delete respectivamente.

```
const router= require ('express').Router();
const conexion=require('./config/conexion')
//asignamos las rutas

//get
router.get('/',(req, res)=>{
  res.setHeader('Access-Control-Allow-Origin', '*');
  let sql='select * from tb_equipo'
  conexion.query(sql,(err,rows,fields)=>{
    if(err) throw err;
    else{
      res.json(rows);
    }
  })
})

// get equipo
router.get('/:id',(req, res)=>{
  res.setHeader('Access-Control-Allow-Origin', '*');
  const{id}=req.params
  let sql='select * from tb_equipo where id_equipo=?'
  conexion.query(sql,[id],(err,rows,fields)=>{
    if(err) throw err;
    else{
      res.json(rows);
    }
  })
})
```

```
//post
router.post('/',(req,res)=>{
  res.setHeader('Access-Control-Allow-Origin', '*');
  const{nombre,logo}= req.body
  let sql=`insert into tb_equipo(nombre,logo) values ('${nombre}','${logo}')`;
  conexion.query(sql,(err,rows,fields)=>{
    if(err) throw err;
    else{
      res.json({status: 'equipo agregado'});
    }
  })
})

//delete
router.delete('/:id',(req,res)=>{
  res.setHeader('Access-Control-Allow-Origin', '*');
  const{id}=req.params

  let sql= `delete from tb_equipo where id_equipo='${id}'`;

  conexion.query(sql,(err,rows,fields)=>{
    if(err) throw err;
    else{
      res.json({status: 'equipo eliminado'});
    }
  })
})
```

```

router.put('/:id', (req, res) => {
  res.setHeader('Access-Control-Allow-Origin', '*');
  const {id} = req.params;
  const {nombre, logo} = req.body;

  let sql = `update tb_equipo set
    nombre = '${nombre}',
    logo = '${logo}'
    where id_equipo = '${id}'`;
  conexion.query(sql, (err, rows, fields) => {
    if (err) throw err;
    else {
      res.json({status: 'equipo modificado'});
    }
  })
})

module.exports = router;

```

7. Instalar nodemon con el siguiente comando “**npm install --save-dev nodemon**”
8. Instalar Express y mysql : “**npm install express mysql**”
9. Ejecutar el comando “npm run dev”
 - a. En caso de dar error en mysql
 - i. Ejecutar en mysql workbench: “ALTER USER 'SU_USUARIO'@'localhost' IDENTIFIED WITH mysql_native_password BY 'SU_Contraseña';”
 - ii. “flush privileges;”
 - iii. Ejecutar nuevamente “npm run dev”

```

PS D:\UCuenca\PrograWeb\apiNodejs> npm run dev
npm WARN config global `--global`, `--local` are deprecated. Use `--location=global` instead.

> apinodejs@1.0.0 dev
> nodemon/index.js

[nodemon] 2.0.19
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node /index.js index.js`
Servidor iniciado en el puerto: 3000
Base de datos conectada

```

10. Los servicios web estan funcionando, se puede probar con postman en la siguiente ruta : <http://localhost:3000/api>
11. Dentro de la carpeta donde queramos crear un nuevo proyecto ejecutar “ng new {nombre del proyecto}”
12. A las dos preguntas contestar con “y” y “css”

```

C:\Users\rauls\Documents\angular>ng new proyecto
? Would you like to add Angular routing? Yes
? Which stylesheet format would you like to use? CSS
CREATE proyecto/angular.json (2932 bytes)
CREATE proyecto/package.json (1039 bytes)
CREATE proyecto/README.md (1062 bytes)
CREATE proyecto/tsconfig.json (863 bytes)
CREATE proyecto/.editorconfig (274 bytes)
CREATE proyecto/.gitignore (548 bytes)
CREATE proyecto/.browserslistrc (600 bytes)
CREATE proyecto/karma.conf.js (1425 bytes)
CREATE proyecto/tsconfig.app.json (287 bytes)
CREATE proyecto/tsconfig.spec.json (333 bytes)
CREATE proyecto/.vscode/extensions.json (130 bytes)
CREATE proyecto/.vscode/launch.json (474 bytes)
CREATE proyecto/.vscode/tasks.json (938 bytes)
CREATE proyecto/src/favicon.ico (948 bytes)
CREATE proyecto/src/index.html (294 bytes)
CREATE proyecto/src/main.ts (372 bytes)
CREATE proyecto/src/polyfills.ts (2338 bytes)
CREATE proyecto/src/styles.css (80 bytes)
CREATE proyecto/src/test.ts (749 bytes)
CREATE proyecto/src/assets/.gitkeep (0 bytes)
CREATE proyecto/src/environments/environment.prod.ts (51 bytes)
CREATE proyecto/src/environments/environment.ts (658 bytes)
CREATE proyecto/src/app/app-routing.module.ts (245 bytes)
CREATE proyecto/src/app/app.module.ts (393 bytes)
CREATE proyecto/src/app/app.component.html (23364 bytes)
CREATE proyecto/src/app/app.component.spec.ts (1079 bytes)
CREATE proyecto/src/app/app.component.ts (212 bytes)
CREATE proyecto/src/app/app.component.css (0 bytes)
- Installing packages (npm)...

```

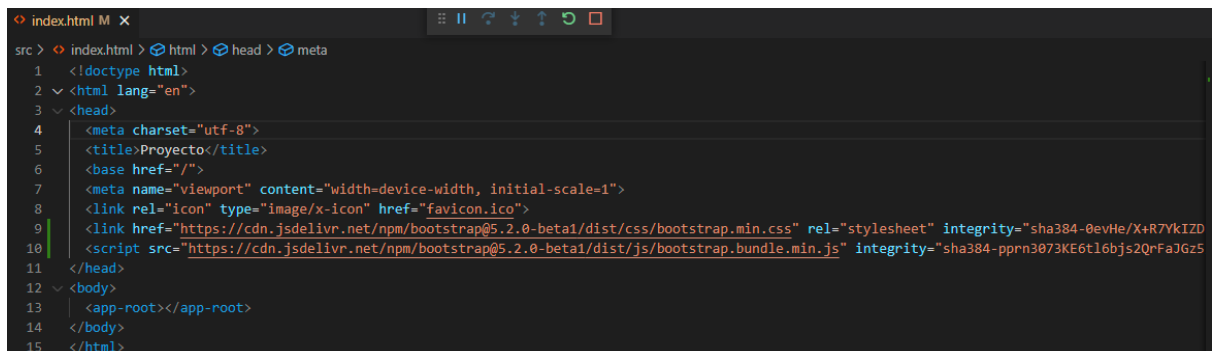
13. Abrir el proyecto con vsCode "code ." dentro del proyecto

14. Añadir las hojas de estilo dentro de src/index.html

```

-<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-beta1/dist/css
/bootstrap.min.css" rel="stylesheet"
integrity="sha384-0evHe/X+R7YkIZDRvuzKMRqM+OrBnVFBL6DOitfPri4tjfh
xaWutUpFmBp4vmVor" crossorigin="anonymous">
-<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-beta1/dist/js/b
ootstrap.bundle.min.js"
integrity="sha384-pprn3073KE6t16bjs2QrFaJGz5/SUSLqktiwsUTF55Jfv3q
YSDhgCecCxMW52nD2" crossorigin="anonymous"></script>

```



```

index.html M X
src > index.html > html > head > meta
1 <!doctype html>
2 <html lang="en">
3 <head>
4   <meta charset="utf-8">
5   <title>Proyecto</title>
6   <base href="/">
7   <meta name="viewport" content="width=device-width, initial-scale=1">
8   <link rel="icon" type="image/x-icon" href="favicon.ico">
9   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-beta1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-0evHe/X+R7YkIZDRvuzKMRqM+OrBnVFBL6DOitfPri4tjfhxaWutUpFmBp4vmVor" crossorigin="anonymous">
10  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-beta1/dist/js/bootstrap.bundle.min.js" integrity="sha384-pprn3073KE6t16bjs2QrFaJGz5/SUSLqktiwsUTF55Jfv3qYSDhgCecCxMW52nD2" crossorigin="anonymous"></script>
11 </head>
12 <body>
13   <app-root></app-root>
14 </body>
15 </html>

```

15. En el terminal(cmd) crear 3 componentes

```
-ng g c COMPONENTES/inicio
-ng g c COMPONENTES/agregar
-ng g c COMPONENTES/modificar
```

```
C:\Users\rauls\Documents\angular\proyecto>ng g c COMPONENTES/inicio
CREATE src/app/COMPONENTES/inicio/inicio.component.html (21 bytes)
CREATE src/app/COMPONENTES/inicio/inicio.component.spec.ts (599 bytes)
CREATE src/app/COMPONENTES/inicio/inicio.component.ts (275 bytes)
CREATE src/app/COMPONENTES/inicio/inicio.component.css (0 bytes)
UPDATE src/app/app.module.ts (487 bytes)

C:\Users\rauls\Documents\angular\proyecto>ng g c COMPONENTES/agregar
CREATE src/app/COMPONENTES/agregar/agregar.component.html (22 bytes)
CREATE src/app/COMPONENTES/agregar/agregar.component.spec.ts (606 bytes)
CREATE src/app/COMPONENTES/agregar/agregar.component.ts (279 bytes)
CREATE src/app/COMPONENTES/agregar/agregar.component.css (0 bytes)
UPDATE src/app/app.module.ts (585 bytes)

C:\Users\rauls\Documents\angular\proyecto>ng g c COMPONENTES/modificar
CREATE src/app/COMPONENTES/modificar/modificar.component.html (24 bytes)
CREATE src/app/COMPONENTES/modificar/modificar.component.spec.ts (620 bytes)
CREATE src/app/COMPONENTES/modificar/modificar.component.ts (287 bytes)
CREATE src/app/COMPONENTES/modificar/modificar.component.css (0 bytes)
UPDATE src/app/app.module.ts (691 bytes)

C:\Users\rauls\Documents\angular\proyecto>
```

16. Modificar el archivo src/app/app.component.html

```
app.component.html M X
src > app > app.component.html > router-outlet
Go to component
1 <nav class="navbar navbar-expand-lg navbar-light bg-light">
2   <div class="container-fluid">
3     <a class="navbar-brand" href="#">CRUD BASICO</a>
4     <div class="collapse navbar-collapse" id="navbarSupportedContent">
5       <ul class="navbar-nav me-auto mb-2 mb-lg-0">
6         <li class="nav-item">
7           <a class="nav-link active" aria-current="page" routerLink="/add">Agregar</a>
8         </li>
9       </ul>
10    </div>
11  </div>
12 </nav>
13
14 <router-outlet></router-outlet>
```

17. Modificar src/app/app-routing.module.ts

```
import { AgregarComponent } from './COMPONENTES/agregar/agregar.component';
import { InicioComponent } from './COMPONENTES/inicio/inicio.component';
import { ModificarComponent } from
'./COMPONENTES/modificar/modificar.component';

{ path:'', redirectTo:'/inicio', pathMatch:'full'},
{path:'inicio' , component: InicioComponent},
{path:'add', component:AgregarComponent},
{path:'edit/:id', component:ModificarComponent}
```

```

TS app-routing.module.ts M X
src > app > TS app-routing.module.ts > AppRoutingModule
2 | import { Routes, RouterModule } from '@angular/router';
3 | import { AgregarComponent } from '../COMPONENTES/agregar/agregar.component';
4 |
5 | import { InicioComponent } from '../COMPONENTES/inicio/inicio.component';
6 | import { ModificarComponent } from '../COMPONENTES/modificar/modificar.component';
7 |
8 | const routes: Routes = [
9 |   { path: '', redirectTo: '/inicio', pathMatch: 'full' },
10 |   { path: 'inicio', component: InicioComponent },
11 |   { path: 'add', component: AgregarComponent },
12 |   { path: 'edit/:id', component: ModificarComponent }
13 | ];
14 |
15 | @NgModule({
16 |   imports: [RouterModule.forRoot(routes)],
17 |   exports: [RouterModule]
18 | })
19 | export class AppRoutingModule { }

```

18. En el terminal crear un nuevo servicio

```

C:\Users\rauls\Documents\angular\proyecto>ng g s /SERVICES/equipo
CREATE src/app/SERVICES/equipo.service.spec.ts (357 bytes)
CREATE src/app/SERVICES/equipo.service.ts (135 bytes)

C:\Users\rauls\Documents\angular\proyecto>

```

19. Modificar src/app/app.module.ts importando las librerías

```

-import { HttpClientModule } from '@angular/common/http';
-import { FormsModule } from '@angular/forms';
-import {
  BrowserModule,
  AppRoutingModule,
  HttpClientModule,
  FormsModule
},

```

```

TS app.module.ts M ●
src > app > TS app.module.ts > ...
1  import { BrowserModule } from '@angular/platform-browser';
2  import { NgModule } from '@angular/core';
3
4  import { AppRoutingModule } from './app-routing.module';
5  import { AppComponent } from './app.component';
6  import { InicioComponent } from './COMPONENTES/inicio/inicio.component';
7  import { AgregarComponent } from './COMPONENTES/agregar/agregar.component';
8  import { ModificarComponent } from './COMPONENTES/modificar/modificar.component';
9
10 import { HttpClientModule } from '@angular/common/http';
11 import { FormsModule } from '@angular/forms';
12
13 @NgModule({
14   declarations: [
15     AppComponent,
16     InicioComponent,
17     AgregarComponent,
18     ModificarComponent
19   ],
20   imports: [
21     BrowserModule,
22     AppRoutingModule,
23     HttpClientModule,
24     FormsModule
25   ],
26   providers: [],
27   bootstrap: [AppComponent]
28 })
29 export class AppModule { }

```

20. Se añaden los servicios rest dentro de src/app/SERVICES/equipo.service.ts
 -import { HttpClient } from '@angular/common/http';

TS equipo.service.ts U X

src > app > SERVICES > TS equipo.service.ts > EquipoService > getUnEquipo

```
1  import { Injectable } from '@angular/core';
2  import { HttpClient } from '@angular/common/http';
3
4  @Injectable({
5    providedIn: 'root'
6  })
7  export class EquipoService {
8
9    url='http://localhost:3000/api';
10    constructor(private http: HttpClient) { }
11
12    //get equipos
13    getEquipos()
14    {
15      return this.http.get(this.url);
16    }
17
18    //get un Equipo
19    getUnEquipo(id:string){
20      return this.http.get(this.url+'/'+id);
21    }
22
23    //agregar equipo
24    addEquipo(equipo:Equipo)
25    {
26      return this.http.post(this.url, equipo);
27    }
28  }
```

```

TS equipo.service.ts U X
src > app > SERVICES > TS equipo.service.ts > EquipoService > getUnEquipo
16     }
17
18     //get un Equipo
19     getUnEquipo(id:string){
20         return this.http.get(this.url+'/'+id);
21     }
22
23     //agregar equipo
24     addEquipo(equipo:Equipo)
25     {
26         return this.http.post(this.url, equipo);
27     }
28
29     //eliminar
30     deleteEquipo(id:string){
31         return this.http.delete(this.url+'/'+id);
32     }
33
34     //modificar equipo
35     editEquipo(id:string, equipo:Equipo){
36         return this.http.put(this.url+'/'+id, equipo);
37     }
38
39 }
40
41 export interface Equipo{
42     id_equipo?:string;
43     nombre?:string;
44     logo?:string;
45 }

```

21. Añadir las funciones con los servicios dentro de
src/app/COMPONENTES/inicio/inicio.component.ts

TS inicio.component.ts U ●

src > app > COMPONENTES > inicio > TS inicio.component.ts > InicioCom

```
17   ngOnInit(): void {
18     this.listarEquipo();
19   }
20
21   listarEquipo()
22   {
23     this.EquipoService.getEquipos().subscribe(
24       res=>{
25         console.log(res);
26         this.ListarEquipo=<any>res;
27       },
28       err => console.log(err)
29     );
30   }
31
32   eliminar(id:string)
33   {
34     this.EquipoService.deleteEquipo(id).subscribe(
35       res=>{
36         console.log('equipo eliminado');
37         this.listarEquipo();
38       },
39       err=> console.log(err)
40     );
41   }
42
43   modificar(id:string){
44     this.router.navigate(['/edit/'+id]);
45   }
46 }
```

```

TS inicio.component.ts U
src > app > COMPONENTES > inicio > TS inicio.component.ts > InicioComponent > ListarEquipo
1  import { Component, OnInit } from '@angular/core';
2  import { EquipoService, Equipo } from '../SERVICES/equipo.service';
3  import { Router } from '@angular/router';
4
5  @Component({
6    selector: 'app-inicio',
7    templateUrl: './inicio.component.html',
8    styleUrls: ['./inicio.component.css']
9  })
10 export class InicioComponent implements OnInit {
11
12     //varibale
13     ListarEquipo:Equipo[];
14
15     constructor(private EquipoService:EquipoService, private router:Router) { }
16
17     ngOnInit(): void {
18         this.listarEquipo();
19     }
20

```

22. Modificar tsconfig.json

```

TS inicio.component.ts U  TS tsconfig.json M X
tsconfig.json > {} compilerOptions
1  /* To learn more about this file see: https://a
2  {
3      "compileOnSave": false,
4      "compilerOptions": {
5          "baseUrl": "./",
6          "outDir": "./dist/out-tsc",
7          "forceConsistentCasingInFileNames": true,
8          "strict": true,
9          "strictPropertyInitialization": false,
10         "noImplicitOverride": true,
11         "noPropertyAccessFromIndexSignature": true,
12         "noImplicitReturns": true,
13         "noFallthroughCasesInSwitch": true,
14         "sourceMap": true,
15         "declaration": false,
16         "downlevelIteration": true,
17         "experimentalDecorators": true,
18         "moduleResolution": "node",

```

23. Crear la configuracion de proxy en la carpeta raíz (proxy.conf.json)

```

{} proxy.conf.json U X
src > {} proxy.conf.json > ...
1  {
2    "/api/*":
3    {
4      "target": "http://localhost:3000",
5      "secure": "false",
6      "logLevel": "debug"
7    }
8  }

```

24. Modificar la config para iniciar el proyecto

```

{} package.json ●
{} package.json > {} scripts
1  {
2    "name": "proyecto",
3    "version": "0.0.0",
4    > Debug
5    "scripts": {
6      "ng": "ng",
7      "start": "ng serve --proxy-conf proxy.conf.json",
8      "build": "ng build",
9      "watch": "ng build --watch --configuration development",
10     "test": "ng test"
11   }

```

25. Configurar el angular.json

```

angular.json
angular.json > {} projects > {} proyecto > {} architect > {} serve > {} options
52     development : {
53         "buildOptimizer": false,
54         "optimization": false,
55         "vendorChunk": true,
56         "extractLicenses": false,
57         "sourceMap": true,
58         "namedChunks": true
59     }
60 },
61     "defaultConfiguration": "production"
62 },
63     "serve": {
64         "builder": "@angular-devkit/build-angular:dev-server",
65         "options": {
66             "proxyConfig": "proxy.conf.json"
67         },
68         "configurations": {
69             "production": {
70                 "browserTarget": "proyecto:build:production"
71             },
72             "development": {
73                 "browserTarget": "proyecto:build:development"
74             }
75         },
76         "defaultConfiguration": "development"

```

26. Modificar src/app/COMPONENTES/inicio/inicio.component.html

```

inicio.component.html
src > app > COMPONENTES > inicio > inicio.component.html > div.container > table.table
1 <div class="container" >
2 <p>LISTA DE EQUIPOS!</p>
3 <button routerLink="/add" type="button" class="btn btn-dark">Agregar </button>
4 <table class="table">
5 <thead>
6 <tr>
7 <th scope="col">Id</th>
8 <th scope="col">Nombre</th>
9 <th scope="col">Logo</th>
10 <th scope="col">Editar</th>
11 <th scope="col">Eliminar</th>
12 </tr>
13 </thead>
14 <tbody>
15 <tr *ngFor="let equipo of ListarEquipo">
16 <th>{{equipo.id_equipo}}</th>
17 <td>{{equipo.nombre}}</td>
18 <td>{{equipo.logo}}</td>
19 <td>
20 <button (click)="modificar(equipo.id_equipo!)" type="button" class="btn btn-primary">Editar</button>
21 </td>
22 <td>
23 <button (click)="eliminar(equipo.id_equipo!)" type="button" class="btn btn-danger">Eliminar</button>
24 </td>
25 </tr>
26 </tbody>
27 </table>
28 </div>

```

27. Modificar src/app/COMPONENTES/agregar/agregar.component.ts

28. Modificar src/app/COMPONENTES/agregar/agregar.component.html

29. Modificar src/app/COMPONENTES/modificar/modificar.component.ts

30.Modificar src/app/COMPONENTES/modificar/modificar.component.html
31.