

Instituto Tecnológico de Costa Rica
Introducción al desarrollo de páginas web
Apunte taller Angular
Alumno: Andrés Sánchez Gómez
Carné: 2017239278

Experiencia con la instalación de la tecnología

Primeramente, para el taller fue requerido instalar Node.js, dado a que es utilizado por Angular y la instalación del CLI (Command Line Interface) es realizada utilizando NPM(Node Package Manager).

Además, para la instalación de las tecnologías se siguió al pie de la letra la guía proporcionada por los compañeros.

Aspectos más relevantes de Angular

Angular fue creado por Misko Herevy, posteriormente fue adoptado por Google y liberado al público.

Características

- Permite crear aplicaciones SPA (Single Page Application), las cuales renderizan elementos dentro de una misma página web según la necesidad de los usuarios y sin tener que redireccionar la página.
- Utiliza Typescript, que es un transpilador encargado de agregar tipos a JavaScript.
- Tiene componentes web, los cuales son creados una vez y pueden ser reutilizados en múltiples páginas utilizando el selector de cada uno de ellos.
- Estructura de archivos organizada, los componentes funcionan similarmente al patrón MVC

Ventajas

- **Two way data binding:** permite a los usuarios intercambiar datos del componente a la vista, y viceversa.
- **Directivas:** permiten modificar elementos del DOM y cambiar su comportamiento. Entre éstas podemos encontrar: Componentes, directivas estructurales y directivas de atributos.
- **Dependency Injection:** patrón de diseño que permite incrementar la modularidad y eficiencia de las aplicaciones. Las dependencias pueden ser servicios u objetos que una clase necesita para funcionar. Estos pueden ser inyectados en múltiples clases para ser utilizados.
- **Documentación:** Angular posee una documentación clara y organizada, así como una amplia comunidad.

Desventajas

- **Rendimiento:** el rendimiento en proyectos grandes puede verse afectado debido a la gran cantidad de componentes que se deben manejar.
- **Curva de aprendizaje:** Angular es un framework que posee una curva de aprendizaje muy alta porque existen muchas maneras de realizar una tarea.

Versiones

- **Angular 1:** Conocido como AngularJS, introduce la creación de aplicaciones SPA(Single Page Applications) y el linkeo dinámico de rutas a subpáginas.
- **Angular 2:** Angular fue reescrito desde cero, produciendo una versión más pequeña del mismo. En esta versión se introducen los componentes para ser inyectados y reutilizados.
- **Angular 3:** Nunca salió al público general por el pánico que existía en el cambio de las versiones.
- **Angular 4:** Mejoramiento en el rendimiento del framework.
- **Angular 5:** Optimización de construcción y compilación de aplicaciones a través del Angular Universal State Transfer.
- **Angular 7:** Optimización de componentes y herramientas.
- **Angular 8:** Disminución del tamaño general de las aplicaciones.

Descripción del taller

El taller propuesto por los compañeros consistió en construir dos componentes. Primero, se creó un formulario para registrar datos básicos de un usuario. Luego, se construyó un componente que permitiera visualizar la información de los usuarios en una tabla.

Algunas imágenes del resultado del taller se muestran a continuación:

Registro de usuarios

Nombre:

Apellidos:

Cedula:

Correo:

Teléfono:

Edad:

Agregar usuario

Listado de usuarios

Nombre	Apellidos	Cédula	Correo	Teléfono	Edad
test	test	test	test	test	test
test2	test2	test2	test2	test2	test2

- app.component.html

```
1 <head>
2   <!-- Latest compiled and minified CSS -->
3   <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
4   <!-- jQuery library -->
5   <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
6   <!-- Latest compiled JavaScript -->
7   <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
8 </head>
9
10 <body>
11   <div class="row">
12     <div class="col-lg-4">
13       <div style="width: 70%; margin: auto;">
14         <app-formulario-registro></app-formulario-registro>
15       </div>
16     </div>
17     <div class="col-lg-4">
18       <div style="width: 70%; margin: auto;">
19         <app-listado-usuarios></app-listado-usuarios>
20       </div>
21     </div>
22   </div>
23 </body>
```

- app.module.ts

```
1 import { BrowserModule } from '@angular/platform-browser';
2 import { NgModule } from '@angular/core';
3 import { FormsModule } from '@angular/forms';
4 import { AppRoutingModuleModule } from './app-routing.module';
5 import { AppComponent } from './app.component';
6 import { UserService } from './user-service.service';
7 import { ListadoUsuariosComponent } from './components/listado-usuarios/listado-usuarios.component';
8 import { FormularioRegistroComponent } from './components/formulario-registro/formulario-registro.component';
9
10 @NgModule({
11   declarations: [
12     AppComponent,
13     FormularioRegistroComponent,
14     ListadoUsuariosComponent
15   ],
16   imports: [
17     BrowserModule,
18     AppRoutingModuleModule,
19     FormsModule
20   ],
21   providers: [UserService],
22   bootstrap: [AppComponent]
23 })
24
25 export class AppModule { }
```

- formulario-registro.component.html

```
1 <form id="form_usuarios">
2   <h1>Registro de usuarios</h1>
3   <h2>Nombre: </h2>
4   <input type="text" class="textfield" name="nombre" [(ngModel)]="usuario.nombre" />
5   <h2>Apellidos: </h2>
6   <input type="text" class="textfield" name="var" [(ngModel)]="usuario.apellidos" />
7   <h2>Cedula: </h2>
8   <input type="text" class="textfield" name="var" [(ngModel)]="usuario.cedula" />
9   <h2>Correo: </h2>
10  <input type="text" class="textfield" name="var" [(ngModel)]="usuario.correo" />
11  <h2>Teléfono: </h2>
12  <input type="text" class="textfield" name="telefono" [(ngModel)]="usuario.telefono" />
13  <h2>Edad: </h2>
14  <input type="text" class="textfield" name="edad" [(ngModel)]="usuario.edad" />
15  <br>
16  <br>
17  <button type="submit" form="form_usuarios" value="Submit" (click)="this.nuevoUsuario()">Agregar usuario</button>
18 </form>
```

- formulario-registro.component.ts

```
1 import { Component, OnInit } from '@angular/core';
2 import { UserService } from '../user-service.service';
3 import { Usuario } from '../usuarios.interface';
4
5 @Component({
6   selector: 'app-formulario-registro',
7   templateUrl: './formulario-registro.component.html',
8   styleUrls: ['./formulario-registro.component.css']
9 })
10 export class FormularioRegistroComponent implements OnInit {
11   usuario: Usuario
12
13   constructor(private userService: UserService) { }
14
15   ngOnInit() {
16     this.usuario = this.userService.nuevoUsuario();
17   }
18   nuevoUsuario(): void {
19     this.userService.agregarUsuarios(this.usuario);
20     this.usuario = this.userService.nuevoUsuario();
21   }
22 }
```

- listado-usuarios.component.ts

```

1  <h1>Listado de usuarios</h1>
2  <div *ngIf="! usuarios.length">
3    <h2>No hay usuarios por el momento</h2>
4  </div>
5  <div>
6    <div *ngIf="usuarios.length">
7      <table>
8        <tr>
9          <td><b>Nombre</b></td>
10         <td><b>Apellidos</b></td>
11         <td><b>Cédula</b></td>
12         <td><b>Correo</b></td>
13         <td><b>Teléfono</b></td>
14         <td><b>Edad</b></td>
15       </tr>
16       <tr *ngFor="let usuario of usuarios">
17         <td>{{usuario.nombre}}</td>
18         <td>{{usuario.apellidos}}</td>
19         <td>{{usuario.cedula}}</td>
20         <td>{{usuario.correo}}</td>
21         <td>{{usuario.telefono}}</td>
22         <td>{{usuario.edad}}</td>
23       </tr>
24     </table>
25   </div>
26 </div>

```

- listado-usuarios.component.ts

```

1  import { Component, OnInit } from '@angular/core';
2  import { UserService } from '../../user-service.service';
3  import { Usuario } from '../../usuarios.interface';
4
5  @Component({
6    selector: 'app-listado-usuarios',
7    templateUrl: './listado-usuarios.component.html',
8    styleUrls: ['./listado-usuarios.component.css']
9  })
10
11  export class ListadoUsuariosComponent implements OnInit {
12
13    usuarios: Usuario[];
14    constructor(private userService: UserService) { }
15    ngOnInit() {
16      this.usuarios = this.userService.getUsuarios();
17    }
18  }

```


- usuarios.interface.ts

```
1  export interface Usuario {  
2      nombre: string;  
3      apellidos: string;  
4      cedula: string;  
5      correo: string;  
6      telefono: string;  
7      edad: string;  
8  }
```

- user-service.service.ts

```
1  import { Injectable } from '@angular/core';  
2  import { Usuario } from '../usuarios.interface'  
3  
4  @Injectable({  
5      providedIn: 'root'  
6  })  
7  
8  export class UserService {  
9      usuarios  
10  
11     constructor() {  
12         this.usuarios = [];  
13     }  
14     getUsuarios() {  
15         return this.usuarios;  
16     }  
17  
18     agregarUsuarios(user: Usuario) {  
19         this.usuarios.push(user);  
20     }  
21  
22     nuevoUsuario(): Usuario {  
23         return {  
24             nombre: '',  
25             apellidos: '',  
26             cedula: '',  
27             correo: '',  
28             telefono: '',  
29             edad: ''  
30         };  
31     }  
32 }
```