

Práctica Grupal: Angular Formularios

1. Crear un nuevo proyecto de Angular.
2. Crear un nuevo componente llamado “componentes/formulario”
3. Modificar el código del archivo “formulario.component.html” con el siguiente:

```
<hr>
<form>

  <div>

    <div class="form-group row">
      <label class="col-2 col-form-label">Nombre</label>
      <div class="col-8">

        <input class="form-control"
          type="text"
          placeholder="Nombre">
      </div>
    </div>

    <div class="form-group row">
      <label class="col-2 col-form-label">Apellido Paterno</label>
      <div class="col-8">

        <input class="form-control"
          type="text"
          placeholder="Apellido Paterno">
      </div>
    </div>

    <div class="form-group row">
      <label class="col-2 col-form-label">Apellido Materno</label>
      <div class="col-8">

        <input class="form-control"
          type="text"
          placeholder="Apellido Materno">
      </div>
    </div>

    <div class="form-group row">
      <label class="col-2 col-form-label">Correo Electrónico</label>
```

```
<div class="col-md-8">

  <input class="form-control"
    type="email"
    placeholder="Correo Electrónico">
</div>
</div>

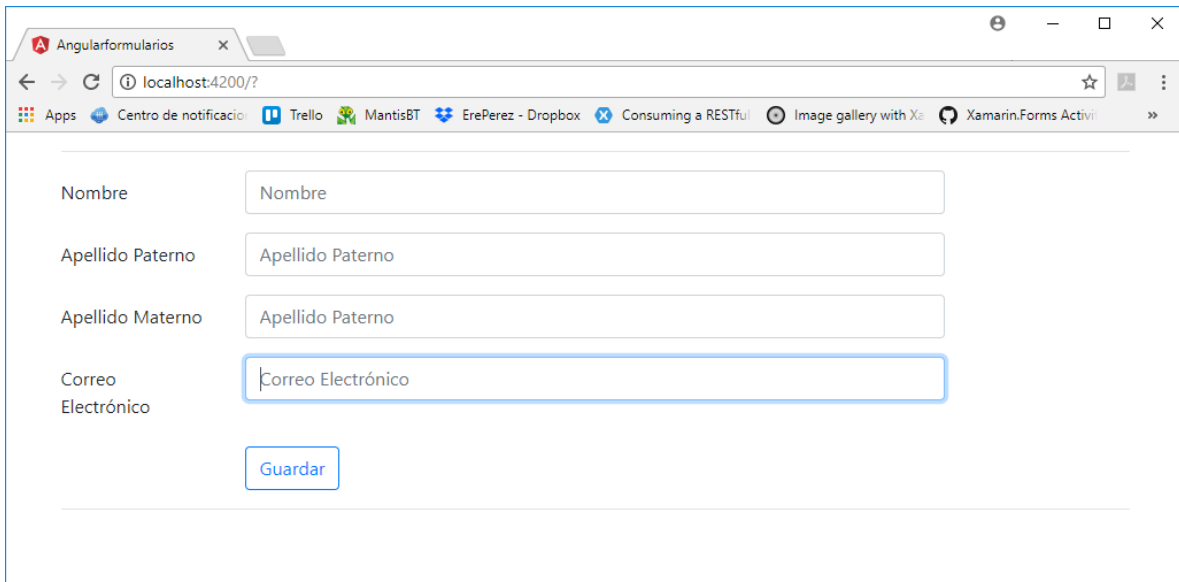
<div class="form-group row">
  <label class="col-2 col-form-label">&nbsp;</label>
  <div class="input-group col-md-8">
    <button type="submit" class="btn btn-outline-primary">
      Guardar
    </button>
  </div>
</div>

</form>
<hr>
```

4. Modificar el código del archivo “app.component.html” con el siguiente:

```
<div class="container main-container">
  <app-formulario></app-formulario>
</div>
```

5. Validar la funcionalidad actual:



Angularformularios x

localhost:4200/?

Nombre

Apellido Paterno

Apellido Materno

Correo Electrónico

6. Modificar el código del archivo “app.module.ts” con el siguiente código:

```
1 import { BrowserModule } from '@angular/platform-browser';
2 import { NgModule } from '@angular/core';
3 import { FormsModule, ReactiveFormsModule } from '@angular/forms';
4
5 import { AppComponent } from './app.component';
6 import { FormularioComponent } from './componentes/formulario/formulario.component';
7
8 @NgModule({
9   declarations: [
10     AppComponent,
11     FormularioComponent
12   ],
13   imports: [
14     BrowserModule,
15     FormsModule,
16     ReactiveFormsModule
17   ],
18   providers: [],
19   bootstrap: [AppComponent]
20 })
21 export class AppModule { }
```

7. Modificar el código del archivo “formulario.component.ts” con el siguiente:

```
1 import { Component, OnInit } from '@angular/core';
2 import { FormGroup, FormControl, Validators } from '@angular/forms';
3
4 @Component({
5   selector: 'app-formulario',
6   templateUrl: './formulario.component.html',
7   styleUrls: ['./formulario.component.css']
8 })
9 export class FormularioComponent implements OnInit {
10
11   formulario: FormGroup;
12
13   constructor() {
14
15     this.formulario = new FormGroup({
16       'nombre': new FormControl(),
17       'apellidoPaterno': new FormControl(),
18       'apellidoMaterno': new FormControl(),
19       'correoElectronico': new FormControl()
20     })
21   }
22
23   ngOnInit() {
24   }
25
26   guardar(){
27     console.log(this.formulario.value);
28   }
29
30 }
31 }
```

8. Modificar el código del archivo “formulario.component.html” con el siguiente:

```
1 <hr>
2 <form [formGroup]="formulario" (ngSubmit)="guardar()">
3
4 <div>
5
6 <div class="form-group row">
```

9. Validar la funcionalidad:

Nombre	<input type="text" value="Nombre"/>
Apellido Paterno	<input type="text" value="Apellido Paterno"/>
Apellido Materno	<input type="text" value="Apellido Paterno"/>
Correo Electrónico	<input type="text" value="Correo Electrónico"/>
	<input type="button" value="Guardar"/>

formulario.component.ts:28

```
▼ {nombre: null, apellidoPaterno: null, apellidoMaterno: null, correoElectronico:
  apellidoMaterno: null
  apellidoPaterno: null
  correoElectronico: null
  nombre: null
  ► __proto__: Object
```

10. Ligamos los campos del formulario con el código, utilizando “formControlName”:

```
<input class="form-control"
  type="text"
  placeholder="Nombre"
  formControlName="nombre">
</div>
```

```
<input class="form-control"
  type="text"
  placeholder="Apellido Paterno"
  formControlName="apellidoPaterno">
</div>
```

```
<input class="form-control"
  type="text"
  placeholder="Apellido Materno"
  formControlName="apellidoMaterno">
</div>
```

```
<input class="form-control"
  type="email"
  placeholder="Correo Electrónico"
  formControlName="correoElectronico">
</div>
```

11. Validar la funcionalidad:

Nombre	<input type="text" value="Jorge"/>
Apellido Paterno	<input type="text" value="Miramontes"/>
Apellido Materno	<input type="text" value="Amezcu"/>
Correo Electrónico	<input type="text" value="jmiramontes@itesm.mx"/>
<input type="button" value="Guardar"/>	

```
formulario.component.ts:28
{nombre: "Jorge", apellidoPaterno: "Miramontes", apellidoMaterno: "Amezcu", correoElectronico: "jmiramontes@itesm.mx"}
  apellidoMaterno: "Amezcu"
  apellidoPaterno: "Miramontes"
  correoElectronico: "jmiramontes@itesm.mx"
  nombre: "Jorge"
  __proto__: Object
```

12. Indicar que todos los campos son requeridos:

```
this.formulario = new FormGroup({
  'nombre': new FormControl('', Validators.required),
  'apellidoPaterno': new FormControl('', Validators.required),
  'apellidoMaterno': new FormControl('', Validators.required),
  'correoElectronico': new FormControl('', Validators.required)
})
```

13. En el archivo “formulario.component.html” se puede utilizar la propiedad “valid” del objeto del formulario para revisar si el formulario es válido en base a las validaciones que se van aplicando:

```
1 <hr>
2 {{ formulario.valid }}
3 <hr>
4 <form [formGroup]="formulario" (ngSubmit)="guardar()">
5
6 <div>
```

14. Validar la funcionalidad:

false

Nombre

Nombre

Apellido Paterno

Apellido Paterno

Apellido Materno

Apellido Materno

Correo Electrónico

Correo Electrónico

Guardar

true

Nombre

Jorge

Apellido Paterno

Miramontes

Apellido Materno

Amezcu

Correo Electrónico

imiramontes@itesm.mx

Guardar

15. Agregar una validación adicional al correo electrónico para indicar que debe cumplir un formato determinado, utilizando expresiones regulares:

```

apellidoMaterno: new FormControl('', validators.required),
'correoElectronico': new FormControl('', [Validators.required, Validators.pattern("[a-z0-9._-]+@[a-z0-9.-]+\.[a-z]{2,3}$")])

```

16. Validar la funcionalidad:

false

Nombre

Jorge

Apellido Paterno

Miramontes

Apellido Materno

Amezcu

Correo Electrónico

imiramontes@

Guardar

17. Agregar una validación adicional al nombre que indique que debe ser requerido y de una longitud mínima de 2 caracteres:

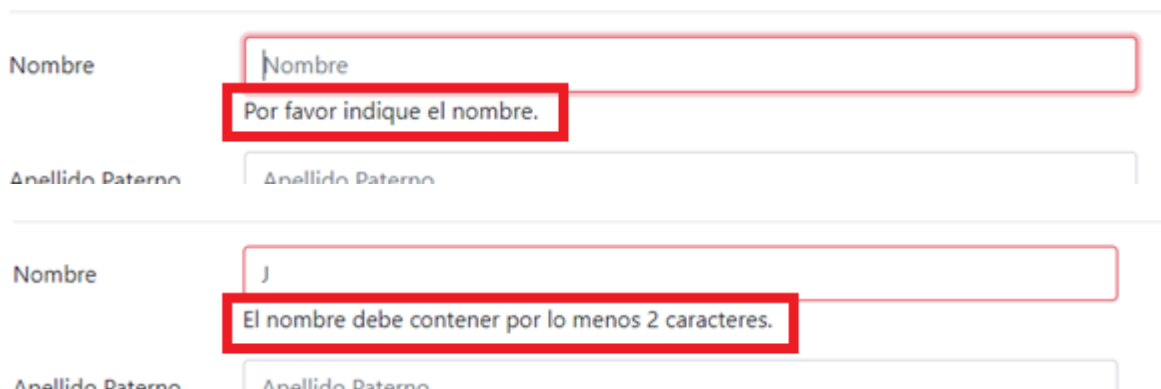
- Modificamos el código del archivo “formulario.component.ts” con el siguiente:

```
this.formulario = new FormGroup({  
  'nombre': new FormControl('', [Validators.required, Validators.minLength(2)]),  
  'apellidoPaterno': new FormControl('', Validators.required),  
});
```

- Modificamos el código del archivo “formulario.component.html” con el siguiente:

```
<form [formGroup]="formulario" (ngSubmit)="guardar()">  
  
  <div>  
  
    <div class="form-group row">  
      <label class="col-2 col-form-label">Nombre</label>  
      <div class="col-8">  
  
        <input class="form-control" [ngClass]="{ 'is-invalid': !formulario.get('nombre').valid }"  
          type="text"  
          placeholder="Nombre"  
          formControlName="nombre">  
  
        <div *ngIf="formulario.controls['nombre'].errors?.required"  
          class="form-control-feedback">  
          Por favor indique el nombre.  
        </div>  
        <div *ngIf="formulario.controls['nombre'].errors?.minlength"  
          class="form-control-feedback">  
          El nombre debe contener por lo menos 2 caracteres.  
        </div>  
  
      </div>  
    </div>  
  </div>  
</form>
```

18. Validamos la funcionalidad:



The screenshot shows two instances of a form. The top instance shows the 'Nombre' field with a red border and a red box around the placeholder text 'Nombre'. Below the field, a red box contains the message 'Por favor indique el nombre.' The bottom instance shows the 'Nombre' field with a red border and a red box around the placeholder text 'J'. Below the field, a red box contains the message 'El nombre debe contener por lo menos 2 caracteres.'

Nombre

Apellido Paterno