Nombre de Postulante: Eduardo Miguel De La Cruz Rojas

Fecha de evaluación: 18/09/2018

Examen Teórico

**Angular**

1. ¿Cómo instalo Angular CLI en mi ordenador?

1. npm install -g angular-cli
2. node install -g angular-cli
3. Angular CLI no se instala, se llama cómo cualquier otro script a través de un tag “script” en mi página

Rpta: a (en la actualidad el modo de instalación es npm install -g @angular/cli)

2. ¿Cómo sirvo/ejecuto una aplicación en Angular 2, gracias a su servidor web integrado?

1. angular-cli serve
2. angular serve
3. node serve
4. ng serve

Rpta: d

3. Se podría decir que un componente es…

1. una directiva con un template.
2. una template con una directiva
3. un elemento del DOM
4. lo mismo que cualquier directiva

Rpta: a

4. ¿Qué comando de Angular CLI debo utilizar para crear un componente llamado mi-componente?

1. ng make component mi-componente
2. ng extend component mi-componente
3. ng create component mi-componente
4. ng generate component mi-componente

Rpta: d

5. ¿A qué se le llama un elemento “host” o anfitrión?

1. Al elemento padre de toda nuestra aplicación Angular
2. A dónde se guardan los datos de nuestra aplicación
3. Al elemento DOM con que se asocia un componente para renderizarse en el DOM.

Rpta: c

6. La API privada de un componente está compuesta por:

1. Propiedades Input (entrada) y output (salida).
2. Sus “lifecycle hooks” (ciclo de vida) y su interacción con el elemento anfitrión (host).
3. Un componente no tiene nada parecido a una API privada.

Rpta: b

7. ¿Esto es correcto? “Para definir propiedades de tipo “input” o entrada se utilizan paréntesis rectos “[]”. Mientras que para las de tipo “output” o salida se utilizan los paréntesis curvos “()”.”

1. No
2. Si

Rpta: b

8. ¿Cuál es la diferencia entre un componente y una directiva?

1. Las directivas son piezas de negocio, mientras que los componentes se suelen usar para presentación y problemas estructurales.
2. Los componentes son piezas de negocio, mientras que las directivas se suelen usar para presentación y problemas estructurales.
3. Son prácticamente lo mismo.

Rpta: b

9. ¿Para qué sirve un decorador?

1. Para extender una función mediante otra función, pero sin tocar la original, que se está extendiendo.
2. Para modificar CSS mediante funciones JS.
3. Para modificar HTML mediante funciones JS.

Rpta: a

10. ¿Qué propiedades funcionan como la API pública de un componente en Angular 2?

1. Input (entrada) y Output (salida)
2. Input (entrada) y Source (fuente)
3. Input (entrada) y Scope (alcance)
4. Input (entrada) y Type (tipo)

Rpta: a

11. ¿En qué carpeta de un proyecto en Angular 2 va nuestro código fuente?

1. dist
2. main
3. src
4. public

Rpta: c

12. Completa la frase correctamente: “Typescript es…”

1. lo que interpreta el navegador.
2. un lenguaje para el desarrollador. El TypeScript compiler se encarga de hacer la conversión de .ts a un .js
3. un lenguaje para qué el desarrollador integre distintas tipografías.

Rpta: b

Examen Práctico

**Typescript**

1) Se requiere que sea lo más específico posible en cuanto a los tipos, sea lo más explícito posible y siga las instrucciones que se encuentran en los comentarios del mismo.

Crear un archivo con nombre prueba\_typescript.ts

Convertir el siguiene Javascript en Typescript

// Uso de Let y Const

var nombre = "César Tapia";

var edad = 32;

var PERSONAJE = {

nombre: nombre,

edad: edad

};

// Cree una interfaz que sirva para validar el siguiente objeto

var batman = {

nombre: "Bruno Díaz",

artesMarciales: ["Karate","Aikido","Wing Chun","Jiu-Jitsu"]

}

// Convertir esta función a una función de flecha

function resultadoDoble( a, b ){

return (a + b) \* 2

}

**Angular**

2) Escriba los comandos para crear un proyecto con nombre prueba-angular, un componente Servicios y un servicio llamado Servicios con angular-cli

$ ng new prueba-angular --style scss

$ ng generate component servicios

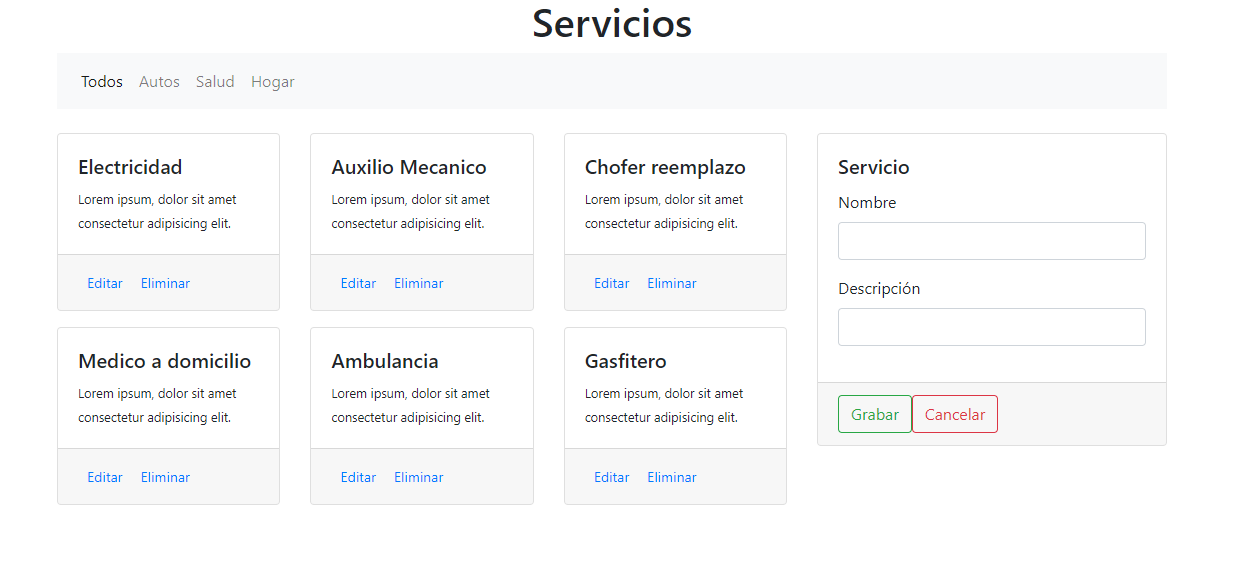
$ ng generate service servicios

3) Realizar el siguiente proyecto:

Listado

Filtros

Formulario Edición



Funciones que debe realizar el aplicativo:

* Listar
* Filtrar
* Editar (mostrar y guardar cambios)

Recuerda que postulas para el perfil de **FrontEnd Developer** el cual demanda un nivel de exigencia alto.   
Está de más decir que se espera una alta calidad de código, patrones de desarrollo y diseño, mantenibilidad, etc.   
Entienda que el ejercicio va a ser revisado por un equipo técnico. Por lo que se le solicita, no deje ningún tópico inconcluso o como supuesto.

**Subir el código a un repo público en GitHub, y enviarnos el link.**

Correos:

**cesartapia@multiplica.com**

**elisairet@multiplica.com**