

PRUEBA DESAROLLADOR ANGULAR

Nombre: Aldair Mosquera Murillo
CC: 1000440313

Sea claro y puntual con sus respuestas. No se extienda demasiado. Si desconoce la respuesta a una pregunta es preferible que se abstenga de responder. Si tiene dudas o sugerencias cópialas al respaldo de esta prueba.

1. Preguntas generales:

1. ¿Qué es responsive design?

Es la capacidad de las páginas web o aplicaciones para adaptarse a diferentes resoluciones de pantalla.

2. ¿Cuál es la diferencia entre desarrollo híbrido, desarrollo nativo y una web app responsive?

Desarrollo híbrido: se crean aplicaciones que están pensadas para funcionar en diferentes entornos; por ejemplo, Angular e Ionic.

Desarrollo nativo: en este caso, las aplicaciones están diseñadas para un entorno específico. Por ejemplo, para desarrollar aplicaciones en Android normalmente utilizamos Java, y en entornos Apple se suele utilizar Swift como lenguaje nativo.

Web app responsive: es una aplicación que tiene la capacidad de adaptarse a diferentes resoluciones de pantalla.

3. ¿Cuáles son los retos al desarrollar una aplicación móvil versus una de escritorio?

Uno de los retos más grandes es el enfoque que se le da a cada entorno. Los usuarios de aplicaciones de escritorio normalmente tienen pantallas más grandes, teclado y ratón; los usuarios de aplicaciones móviles no cuentan con estas ventajas, por lo tanto, la interacción del usuario con la app cambia según el dispositivo.

En escritorio, la información suele ser más completa porque hay más espacio; en móvil se presenta menos información. Las interacciones también deben adaptarse a cada entorno.

Sin embargo, estos no son los únicos elementos a tener en cuenta: también es importante la gestión de recursos (memoria, multimedia y servicios).

4. ¿Cuáles son las principales diferencias entre Javascript y Typescript?

JavaScript es el lenguaje; TypeScript es una capa que le agrega al lenguaje la capacidad de tener tipado.

2. Preguntas Angular

1. ¿Qué es angular?

Es un framework creado por el equipo de Google para crear aplicaciones web.

2. ¿Qué son los componentes en Angular?

Un componente es la forma en que Angular agrupa el código HTML, CSS y TypeScript. Podemos entenderlo como una pieza de Lego dentro del ecosistema de una aplicación.

3. ¿Qué es inyección de dependencias?

Es la capacidad de utilizar una clase o servicio reutilizando su instancia. Esto quiere decir que la clase o el servicio se invoca una sola vez y luego puede reutilizarse para gestionar mejor la memoria de la aplicación.

4. ¿Qué es un pipe y para qué sirve?

Un pipe o tubería es una característica de Angular que nos permite transformar información en un componente. Es decir, un pipe puede utilizarse para dar formato a los datos (fechas, moneda, texto, etc.).

5. ¿Qué es un template y para qué sirve?

un template es la forma resumida de un componente

6. ¿Qué es una directiva y para qué sirve?

Las directivas son una de las formas que utiliza Angular para procesar y definir comportamientos en la información antes de ser presentada al usuario.

7. ¿Qué son los interceptores en Angular?

Estos nos permiten interceptar peticiones HTTP y manipularlas antes de que sean ejecutadas. Por ejemplo, se pueden utilizar interceptores para agregar cabeceras de seguridad a una petición.

8. ¿Cuáles son los ciclos de vida en angular?

9. ¿Cuáles son las diferencias entre Constructores y OnInit?

El constructor es la primera función que se invoca cuando se crea el componente; OnInit se ejecuta cuando el componente ya ha sido inicializado.

10. ¿Cómo se hace uso correcto de ngOnInit() en una clase?

Lo explico con un ejemplo: si tengo un formulario y necesito inicializarlo con los valores de un servicio, podría hacerlo desde ngOnInit, ya que este garantiza que el formulario esté renderizado antes de actualizarlo con la información del servicio.

11. ¿Qué son @Inputs y @Outputs?

Los @Input le permiten a un componente recibir propiedades de componentes externos; los @Output hacen algo similar, pero exponen eventos que pueden ser emitidos y escuchados por otros componentes.

12. Describa brevemente para que se utiliza: {{}} en un template.

Esto se utiliza para interpolar información.

Ejemplo: <p>¿Cuánto es 4 + 3? = {{ 4 + 3 }}</p>

Esto mostraría: "¿Cuánto es 4 + 3? = 7"

13. Describa brevemente para que se utiliza: [(ngModel)]... en un template.

el [(ngModel)] o el modelo banana es lo que le permite a Angular manipular información de los ts y el html en tiempo real

3. Preguntas Ionic

14. ¿Cuál es la funcionalidad del archivo src/app/app.module.ts?

15. ¿Cuál es el punto de entrada de una aplicación (nombre del ts y el html)?

16. Escriba el comando necesario para probar una aplicación en un dispositivo real utilizando livereload.

17. ¿Cuál es el lenguaje de hojas de estilo utilizado por default?

18. ¿Cuál es el lenguaje de programación utilizado por default?

19. ¿Cuáles plataformas son soportadas actualmente por Ionic?

20. ¿Existe la forma de dar estilos individualmente para cada plataforma? ¿O se deben ajustar todas las plataformas a un único estilo?

21. ¿Cuáles de los siguientes decoradores pueden ser utilizados en una aplicación?

- a) @Directive
- b) @Component
- c) @Target
- d) @NgModule
- e) @Pipe
- f) @Injectable
- g) @BackgroundTask
- h) @HttpRequest

22. ¿Qué son Promises y Observables?

23. Diga que significa cada línea en el siguiente fragmento de código del archivo src/app/app.module.ts:

```
//...
@NgModule({
  declarations: []]
```

```
imports: [],
providers: [],
bootstrap: []
//...
```

24. Diga que significa cada línea en el siguiente fragmento de código:

```
//...
```

```
import {DisponibilidadAprobacionService} from
'./disponibilidad-aprobacion.service';
```

```
@Component({
  selector: 'rubros',
  templateUrl: 'rubro.view.html'
})
```

```
export class RubroComponent implements OnInit
{
  @Input() params;

  constructor(public service: DisponibilidadAprobacionService)
  {}

  ngOnInit()
  {
    //...
  }
//...
```

25. ¿Cómo se logra manipular el hardware de un dispositivo desde Ionic?

26. Mencione ventajas y desventajas de Ionic.