

# Instituto Tecnológico de Culiacán



## Ingeniería en tecnologías de la información y comunicación

**“Tarea: Preguntas sobre el framework Angular”**

**Materia:** Programación Web

**Nombre del alumno:** Jorge Osuna Quintana

**Maestro:** M.C. Francisco González Hernández

**Fecha de entrega:** 21 de abril de 2020

## ¿En cuál(es) problemáticas se enfoca Angular?

Angular como framework de TypeScript enfoca su filosofía de acuerdo a un paradigma enfocado al diseño, escritura, lectura y uso de aplicaciones Web para plataformas de escritorio y móviles que promuevan la escalabilidad del software.

En el desarrollo de software por lo general hacemos uso librerías o módulos nativos del lenguaje en el que programamos debido a que consideramos fiable el utilizar funciones built-in o rutinas incluidas en estos módulos debido a que cumplen (por lo general) de manera eficiente con las necesidades del software que estamos creando. A pesar de esto último, estas rutinas no responden a un método ordenado de trabajo ni a una metodología que nos permita entender la manera en que debemos de crear nuestro producto. Todos estos factores resultan en complicaciones como bien puede ser:

- El código se vuelve prácticamente ilegible a lo largo del tiempo de desarrollo.
- Debido a que no seguimos metodologías adecuadas, el código tiende a volverse no-mantenible y las distintas partes que lo conforman se vuelven completamente dependientes entre sí.
- La cantidad de espacio (líneas de código) que utilizamos para crear funciones o partes de nuestro software significan una porción significativa de nuestra aplicación.
- Debido a los distintas partes que componen una aplicación Web, se necesita interactuar constantemente con componentes que no corresponden propiamente al componente que estamos programando lo cual implica en distintas ocasiones una conversión de tipo de datos que implican ciertos recursos (por ejemplo: La creación de HTML a través de cadenas de texto que se castean hacia el front-end desde el back-end).

## ¿Cuáles son sus propuestas de solución?

Para entender cómo el framework soluciona los problemas presentados en la sección anterior debemos entender que Angular basa sus operaciones de acuerdo al MVC (Model View Controller) o Modelo-Vista-Controlador. La idea central detrás de **MVC** es que permite la separación clara del código entre el manejo de los datos (**modelo**), la lógica de la aplicación (**controller**) y la manera en que presentamos los datos al usuario (**view**).

Se entiende entonces, que Angular trabaja bajo una arquitectura *basada en componentes*. Estos componentes pueden ser fácilmente reemplazados y desacoplados al mismo tiempo en alguna otra parte de la aplicación. Así mismo, se dice que cada componente trabaja lógica y funcionalmente de forma independiente.

Todo lo anterior trae como consecuencia una legibilidad y coherencia del código y de sus componentes destacable, así como la reusabilidad de código permite reducir considerablemente el tamaño de los archivos fuente de la aplicación y muy recurrentemente, su tiempo de ejecución. Debido a la independencia de las partes de las aplicaciones desarrolladas podemos modificar meramente una parte del código sin afectar enteramente a toda la aplicación.

Otro aspecto a considerar es la interoperabilidad de los datos de la aplicación o el *Data Binding*. A diferencia de código hecho directamente en JavaScript o TypeScript vanilla mediante el DOM en el que se quiere interactuar con la vista de la aplicación, Angular no requiere crear funciones para la generación de elementos HTML porque ya cuenta con un código de sincronización Modelo-Vista que nos ahorra el trabajo. En cambio, podemos interactuar directamente con el modelo o los datos de la aplicación y Angular actualizará la vista automáticamente.

## BIBLIOGRAFÍA

- Angular University. (2019, Abril 26). *Angular For Beginners Guide: Why Angular? Understanding The Top Benefits*. Recuperado de: <https://blog.angular-university.io/why-angular-angular-vs-jquery-a-beginner-friendly-explanation-on-the-advantages-of-angular-and-mvc/>
- Harkushko L. (s.f.). *Angular: Best Use Cases and Reasons To Opt For This Tool*. Recuperado de: <https://yalantis.com/blog/when-to-use-angular/>.
- Green, B., & Seshadri, S. (2013). *AngularJS*. " O'Reilly Media, Inc."