

## DIAGRAMA DE COMPONENTES FRONTEND (ANGULAR)

A continuación, se muestra el diagrama de componentes para el FrontEnd usando Angular.

Y una descripción de cada módulo que lo componen:

Figura 1. Diagrama de componentes FrontEnd

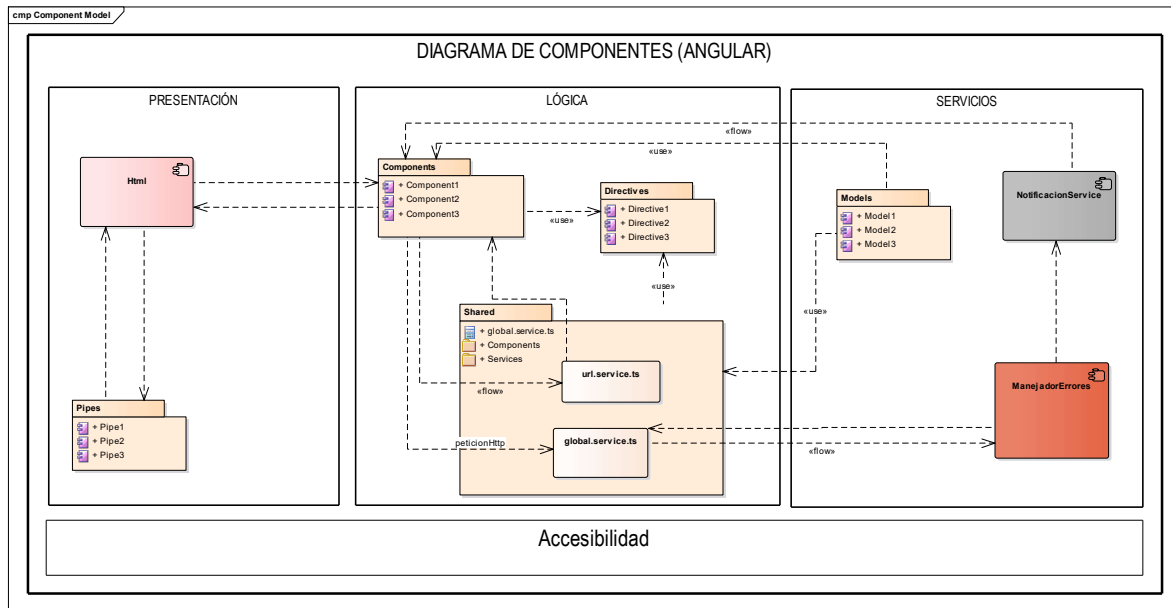


Tabla 1. Descripción diagrama de componentes

ELEMENTO	DESCRIPCIÓN
<b>Html</b>	Son todos los componentes visuales (páginas html) de cada componente en las que se muestra la información en cada pantalla
<b>Accesibilidad</b>	Funcionalidad transversal que controla la configuración para manejo de usabilidad y accesibilidad de cada página web. Controla la forma visual en la que se muestra la información en pantalla y se almacenan en una tabla de la DB para mantener la configuración de esta por cada usuario dentro del App
<b>Pipes</b>	Son componentes que se inyectan directamente en las páginas Html de cada componente para mostrar la información de una manera diferente, pero manteniendo la fuente de información igual. Ej. uso de

	camelCase en nombres, transformación de fuentes, estilos, etc....
<b>Components</b>	Controlan la lógica de cada página, reciben la información que posteriormente se muestra en la pantalla
<b>Shared</b>	Es una carpeta genérica dónde se almacenan los componentes, páginas y servicios como utilitarios o constantes para ser usadas en cualquier lugar o componente dentro del proyecto. Son clases que se crean manualmente
<b>Directives</b>	Son funciones que permiten controlar el comportamiento de una página, manipularla y reutilizar código HTML en cualquier otra. Ej. Controlar el número o tipo de caracteres en un Input
<b>Models</b>	Carpeta que contiene todos los modelos (DTO's, Interfaces) con las propiedades de cada objeto a usar dentro del aplicativo
<b>ManejadorErrores</b>	Clase que controla los diferentes tipos de errores que se presenten al consumir algún servicio REST
<b>NotificacionService</b>	Clase que muestra en pantalla diferentes tipos de mensajes dependiendo el tipo de error que se genere en la clase de manejadorErrores
<b>url.service.ts</b>	Clase en donde se declaran las rutas base de petición de los servicios al backend, allí se encuentran los métodos encargados de generar las url con los parámetros y filtros que requiere cada servicio. Se llama desde cada componente donde se requiera utilizar un llamado http.
<b>Global.service.ts</b>	Clase en donde se encuentran los métodos genéricos de peticiones HTTP (GET, POST, PUT, DELETE), con la cabecera requerida por el API.