PROGRAMACIÓN HÍBRIDA

EXAMEN N° 1 “Fundamentos de IONIC”

Diego Mena - Analista Programador - 2024

INFORME DE DESARROLLO

1. Se crea proyecto en Visual Studio Code usando Ionic + Angular para plataforma Web. Seleccionamos plantilla en blanco.
2. Creamos una nueva carpeta “modelo” dentro del proyecto (src/app/modelo).

Dentro de la misma creamos un archivo figuras.ts (Typescript) para programar las clases que vamos a utilizar en el proyecto. Para su desarrollo nos guiamos atreves del diagrama de clases en UML proporcionado, teniendo en cuenta su jerarquía y herencia.

Cada una con la propiedad, constructor y método para calcular el perímetro de la figura correspondiente.

1. Utilizamos el plugin de Ionic para crear dos componentes: círculo y triángulo.

Para el código de la plantilla de ambos componentes nos apoyamos en la documentación y recursos de Ionic como ion-card, ion-select, ion-text, ion-button, ion-item, ion-list.

Para el código de la clase de ambos componentes se importan los componentes necesarios de Angular y Ionic.

Se importan las clases creadas anteriormente (figuras.ts)

Se usan propiedades para guardar los valores de las medidas ingresadas y el cálculo del perímetro de la figura correspondiente.

Se usan métodos para calcular el perímetro. Éste comienza cuando se hace click en el botón con el evento asociado.

Se convierten los valores a numero entero, se crean instancias de la clase correspondiente enviando el valor como parámetro al constructor.

El método que calcula el perímetro de la figura abstracta (figuras.ts) devuelve el valor del perímetro a la figura correspondiente y se redondea el resultado a dos decimales. Finalmente se actualiza la propiedad “resultado” con el valor del cálculo del perímetro.

1. Para integrar los componentes creados en el home, se tiene en cuanta que:
   1. Todos los códigos (TS) de los componentes y el home se declare “standalone” en modo “true”.
   2. Llamar a todos los componentes y módulos utilizados de Ionic,
   3. las clases (codificadas en figuras.ts) a través del comando “import” y también declaradas en “@Component” por medio de “import”.
   4. También se agrega un evento para poder seleccionar los distintos tipos de figuras a través de $event.detail.value
   5. En el archivo html del home en “ion-toolbar” se usan “ion-title” para el titulo principal y también se incluye “ion-button slot=”end” para agregar el menú de las distintas figuras a seleccionar del lado izquierdo del toolbar.
   6. Se utiliza “ion-change” para manejar la selección de las figuras.
   7. En el “content” Se utiliza la directiva \*ngIf para mostrar el elemento correspondiente cuando éste se encuentre en modo “true”.
   8. Al final se utiliza “ion-row” para agregar una línea de color azul al toolbar.
2. Surgieron dificultades para integrar la imagen en cada uno de los componentes, se trató de utilizar “ion-image” para su implementación, pero no se logró. Después de varios intentos de abordar el problema se decide agregar la etiqueta de html <img /> y finalmente ésta si funcionó.

Cabe destacar como dato indispensable que, para integrar las imágenes dentro del proyecto y que puedan ser visualizadas correctamente deben estar agregadas dentro de la carpeta (src/app/**assets**) de lo contrario al ejecutar la aplicación no se visualiza y en su lugar aparece un icono de un archivo de imagen rota.

1. En conclusión, creo que Ionic + Angular ofrecen muchas herramientas muy útiles e interesantes para el desarrollo de aplicaciones hibridas, destacando la versatilidad y beneficio que ofrece desarrollar código que puede ser ejecutado en distintas plataformas y sistemas sin tener que por ello realizar código por separado para cada una de ellas.

Personalmente el único camino para el buen dominio y poder sacarle el mayor partido a estos excelentes frameworks radica en realizar “mucha practica” e “indagar en la documentación oficial” para ir conociendo mas herramientas y así poder enfrentarse al desarrollo de un código cada vez más completo y complejo para enfrentarse a mejores soluciones informáticas.