Memoria de Figuras geométricas

José Luis Fernández Pérez, 29 de mayo de 2015.

Contenido

[1. Análisis de requisitos 2](#_Toc420568242)

[2. Diseño 2](#_Toc420568243)

[a. Diagrama de clases 3](#_Toc420568244)

[b. Diagrama de paquetes 4](#_Toc420568245)

[3. Código 5](#_Toc420568246)

[4. Documentación: 5](#_Toc420568247)

# Análisis de requisitos

Realizar un programa que, calcule y devuelva por pantalla el valor del área y el volumen de los siguientes poliedros regulares:

* Cubo
* Tetaedro
* Icosaedro
* Octaedro
* Dodecaedro
* Esfera

Para ello se opta por que el usuario elija en un menú en modo texto que Figura en 3D regular quiere calcular, introduciendo la longitud de sus lados para los poliedros o su radio en las esferas.

Adicionalmente se mostrará por interfaz gráfica polígonos regulares 3D. La forma de introducir los datos de esos polígonos será mediante

# Diseño

Para desarrollar esta práctica se utilizará el lenguaje orientada a objetos Java 1.6 mediante el entorno Eclipse Luna

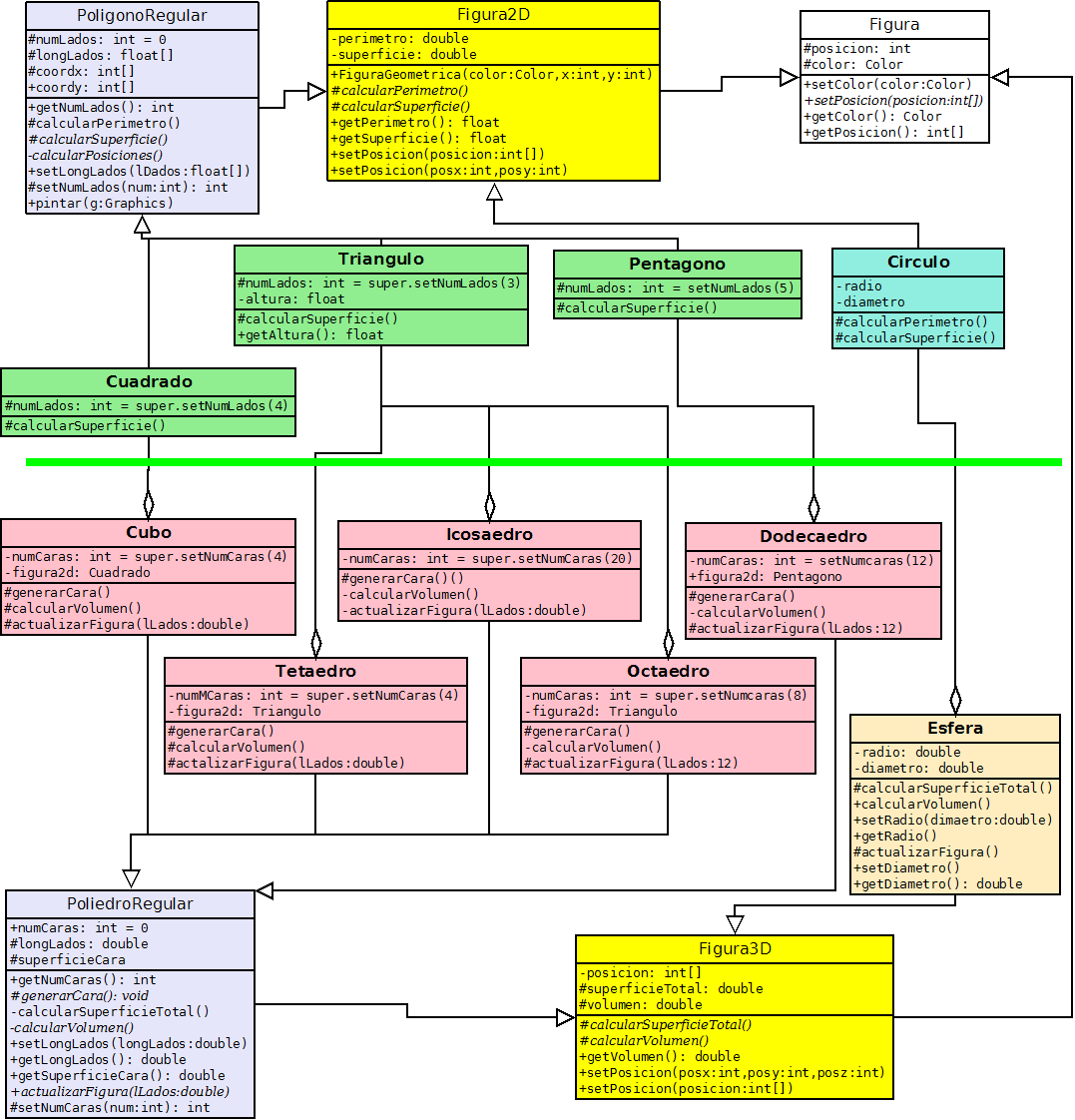
La implementación del menú en modo texto se utiliza una clase previamente creada para otros casos prácticos llamada Utils Console que se encuentra liberada en GitHub[[1]](#footnote-1). Sin embargo es necesario incorporar una nueva función: obtención de una opción introducida mediante texto.

Cada una de las figuras geométricas de 3 dimensiones en la realidad están compuestas por varias figuras geométricas de 2 dimensiones, por lo que se representarán en las clases creadas mediante relaciones de composición (se instancian objetos de 2 dimensiones dentro de las clases de 3 dimensiones). La relación de composición solamente se implementará en los poliedros, que están compuestos por polígonos. En el caso de la esfera, la relación de estar formada por varios círculos se omitirá ya que da complejidad innecesaria al código.

Al ser polígonos regulares, no hay necesidad real de calcular de forma independiente cada una de las caras, ya que cada polígono será igual a los otros (por lo que solo se instanciará un polígono por cada poliedro).

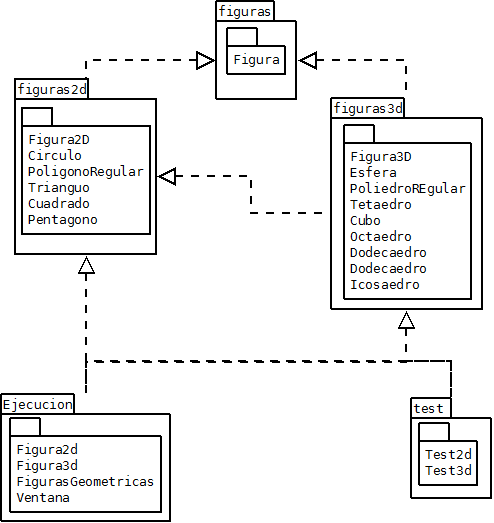
Se entienden que todas las figuras geométricas de 2D y 3D tienen algo en común: son figuras con una posición y un color. Gran parte de las figuras en 2D son Polígonos Regulares con un comportamiento similar. Lo mismo ocurre con las figuras 3D, que casi todas son Poliedro Regulares y por tanto con un comportamiento similar. Como se puede ver en el diagrama de clases, estas relaciones se reflejan en la herencia.

## Diagrama de clases

Leyenda: las clases que su nombre no está en negrita son abstractas. Las funciones y métodos que están en cursiva son abstractas.

Las clases representadas en el diagrama no son las únicas existentes, hay más (las utilizadas para la interfaz de usuarios y test), sin embargo, no utilizan realmente programación orientada a objetos por lo que se omiten del diagrama de clases, no así del diagrama de paquetes.

## Diagrama de paquetes



En cuanto al paquete de ejecución, las distintas clases sirven para:

* FigurasGeometricas: lanza el hilo de ejecución y tiene los menús. Construye la interfaz de consola y llama a la interfaz gráfica.
* Figuras2d: calcula figuras en 2D y devuelve datos sobre ellas.
* Figuras3d: calcula figuras en 3D y devuelve datos sobre ellas.
* Ventana: genera la interfaz gráfica de usuario y contiene una lista de todas las instancias de las figuras en 2D.

# Código

El código puede ser consultado en el repositorio Figuras de Github. Las librerías requeridas se encuentran en el repositorio Utils.

# Documentación:

Se puede encontrar documentación interna con javadoc y documentación externa en el directorio *doc*.

<https://github.com/jfernandezpe/Figuras/tree/master/doc/docManual>

<https://github.com/jfernandezpe/Figuras/tree/master/doc/javadoc>

1. https://github.com/jfernandezpe/Utils [↑](#footnote-ref-1)