In this project we have two basic general type of interfaces named Shape and Visitor. Shape has only one method signature ‘accept()’, which allows a visitor to visit any particular kind of object that implements Shape. Visitor also contains some method signatures which represents functions from a basic Shape to a generic Result type.

We have a composite class named Group which consists of a combination of basic Shape instances, like, Circle, Rectangle, Polygon etc. Polygon again consists of 2 or more basic Point object.

And we have some other decorator classes, like, Stroke, Fill, Outline and Location. Stroke sets the foreground color for drawing any particular Shape. Fill draws a filled Shape instead of an outlined one. Outline draws an outlined Shape instead of an filled one. And Location only specifies a Shape’s location. All of these decorator classes implements the basic Shape interface.

We have also two visitor classes, BoundingBox and Size. BoundingBox is a visitor who visits each Shape and draws the smallest rectangle containing that Shape. Size computes the number of basic Shape’s exists within a Shape(possibly complex). Both of this two classes implements the Visitor interface. We have another visitor class ‘Draw’ which draws a Shape to an android canvas.

We have a test fixture class named ‘Fixtures’, which creates all the objects that we need to draw and send them to the Android Canvas.