You are tasked with developing a mobile application that allows users to create and customize shapes of various types (e.g., circles, squares, triangles) and colors (e.g., red, blue, green). To implement this, you decide to create classes for different shapes and colors.

1. Define a base class called `Shape` with the following properties and methods:
    - Properties: `name` (string) - representing the name of the shape.
    - Methods:
        - `area()` - returns the area of the shape.
        - `perimeter()` - returns the perimeter of the shape.
2. Implement three subclasses of `Shape`:
    - `Circle`: This class should have a property `radius` (float) and should override the `area()` and `perimeter()` methods accordingly.
    - `Square`: This class should have a property `side_length` (float) and should override the `area()` and `perimeter()` methods accordingly.
    - `Triangle`: This class should have properties `side1`, `side2`, and `side3` (float) representing the lengths of the triangle's sides. Override the `area()` and `perimeter()` methods accordingly.
3. Define a class called `Color` with the following properties:
    - Properties: `name` (string) - representing the name of the color.
    - Methods: `showcolor()`
4. Implement three subclasses of `Color`:
    - `Red`, `Blue`, and `Green`: Each subclass should simply inherit from `Color` and have no additional properties or methods.
5. Create instances of shapes (e.g., circles, squares, triangles) and colors (e.g., red, blue, green) by interacting with necessary UI components from home activity. Add UI components to show information about your created objects.
6. Develop unit tests for each of the shape subclasses (Circle, Square, Triangle) and the Color subclasses (Red, Blue, Green) to ensure that their methods (area(), perimeter()) and properties (name) function correctly.
7. Design UI tests to ensure that the user interface of the application behaves as expected when interacting with shapes and colors.
8. Implement continuous integration practices to automate the testing process.