Preamble

You are expected to create a utility SwiftUI application to convert Distance in meters to other units. The conversions to other units are as follows
1. Meters to feet (1m = 3.28 ft)
2. Meters to yards (1m = 1.09 yd)
3. Meters to kilometers (1m = 0.001 km)
4. Meters to centimeters (1m = 100cm)

Before the assessment begins, do the following:

1. Create a SwiftUI application named DistanceConverter
2. Ensure the project runs.
3. Go to the website named Lorem Picsum and download an image with dimensions 300x300 px (https://picsum.photos/300/300)
4. Import the image into the project by dragging the file into the Assets catalog (Rename the file as you wish).

Instructions

Build a SwiftUI application that takes the users input in meters and convert it to the other units mentioned above.

- Input
  - The application allows the user to enter values to the textfield
- Output
  - The converted value of meters to a unit selected by the user

- User interface:

Distance Converter

Unit in meters

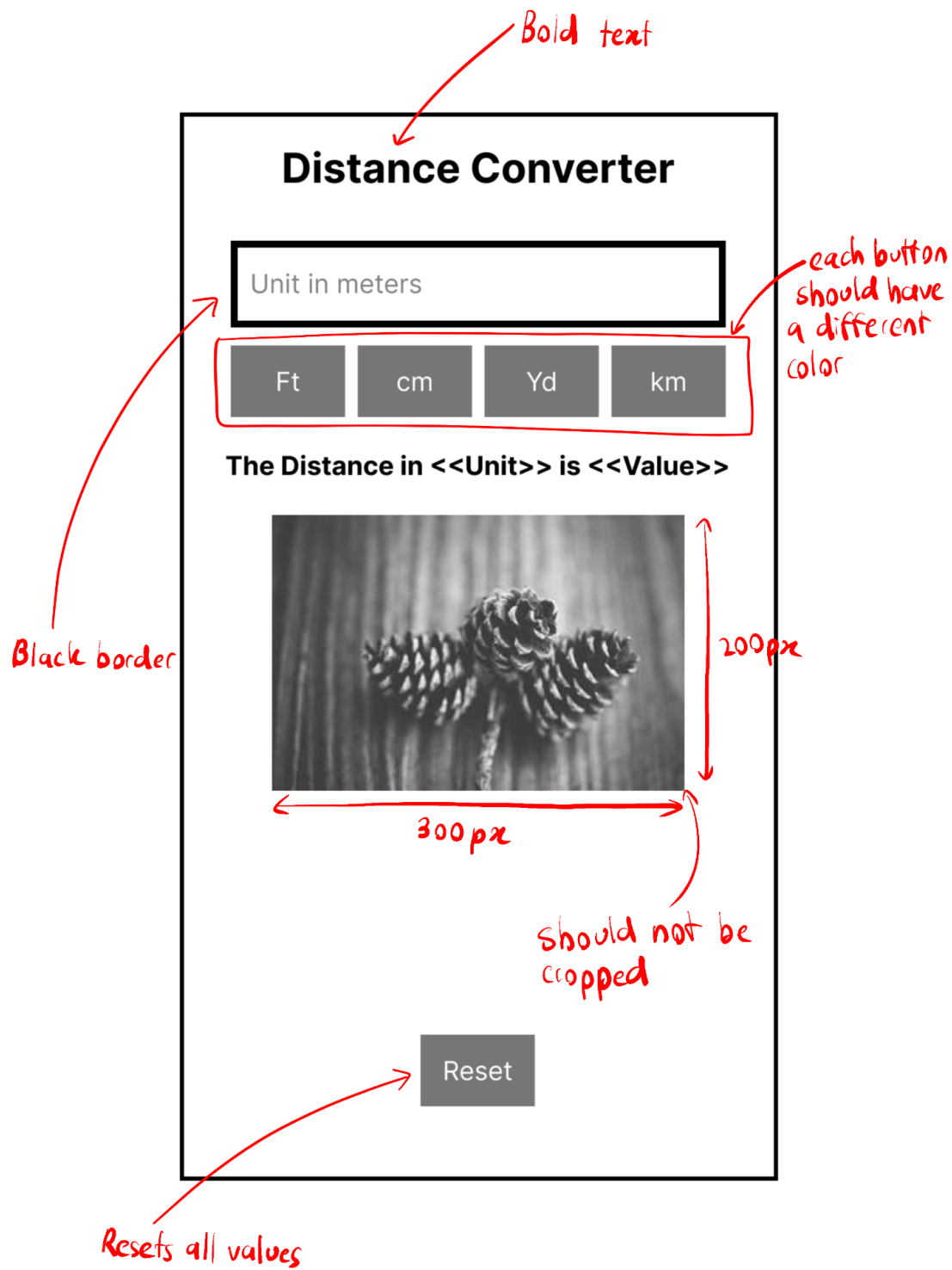| Ft | cm | Yd | km |

The Distance in <<Unit>> is <<Value>>

Reset

- User interface details:

Bold text

Distance Converter

Unit in meters

| Ft | cm | Yd | km |

each button should have a different color

The Distance in <<Unit>> is <<Value>>

Black border

200px

300px

Should not be cropped

Reset

Resets all values

- More details
  - User input should be persistent (You can use @AppStorage to fulfill this purpose)
    - Use case - When the user types "0" in the textfield and terminates/clears the app from memory, once the app is reopened the "0" text should be in the text field
  - The output should be rounded off to 2 decimal places
  - Each of the buttons should be in different colors
  - The image in the UI should not be cropped by any means. It should perfectly fit into the space given.
  - Step into the shoes of the user and see if it's ideal to do something. See how you can avoid the app breaking/crashing.
  - **Ideal, but not necessary** to have the output ONLY shown if and when the user taps on one of the buttons. Till then, it should be hidden.

- **If Errors are encountered**
  - **Check the code for any compile errors (Can be found by building the project - Product -> Build)**
  - **Ensure that a valid iPhone simulator is selected.**
  - **Clean the project (Product -> Clean Build Folder)**
  - **If all else fails, Clean and Restart XCode (It will fix any errors pertaining to XCode 99% of the time)**