

**Aim:** To include icons, images, fonts in Flutter app

**Theory:**

### ***Using Icons in Flutter***

Icons in Flutter can be added using the built-in Material Icons or custom icon packs.

(a) Material Icons

Flutter provides a collection of built-in Material Icons, which can be used with the Icon widget.

Eg: `Icon(Icons.home, size: 30, color: Colors.blue)`

(b) Custom Icons

If you need icons that are not available in the Material Icons set, you can use external icon packs like:

- Font Awesome (font\_awesome\_flutter package)
- Custom SVG Icons (flutter\_svg

package) Eg in pubspec.yaml file -

dependencies:

font\_awesome\_flutter: ^10.5.0

In code -

```
import 'package:font_awesome_flutter/font_awesome_flutter.dart';
```

```
IconButton(  
  icon: Falcon(FontAwesomeIcons.heart, color: Colors.red),  
  onPressed: () {},  
)
```

### ***Adding Images in Flutter***

Images can be loaded in Flutter from different sources like assets, network, or memory.

(a) Using Network Images

Network images are loaded from an online URL. Example:

Eg: `Image.network("https://example.com/sample.jpg", width: 200, height: 150)`

## (b) Using Asset Images

To use images from the local project folder (assets/), follow these steps:

1. Place the image inside the assets/images/ folder.
2. Declare the image in pubspec.yaml:

```
flutter:  
  assets:  
    - assets/images/sample.png
```

In code: Image.asset("assets/images/sample.png", width: 200, height: 150)

## ***Adding Custom Fonts in Flutter***

Custom fonts improve the visual identity of an app.

Steps to Add a Custom Font:

1. Download the font and place it inside the assets/fonts/ folder.
2. Declare the font in pubspec.yaml:

```
flutter:  
  fonts:  
    - family: CustomFont  
      fonts:  
        - asset: assets/fonts/CustomFont-Regular.ttf  
        - asset: assets/fonts/CustomFont-Bold.ttf  
        weight: 700
```

In code

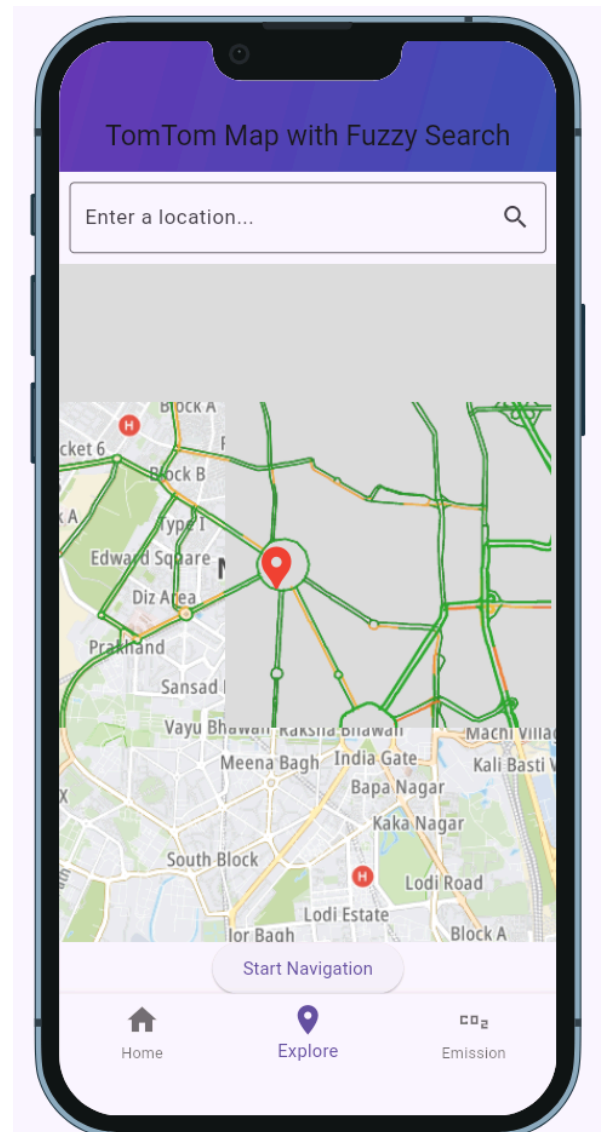
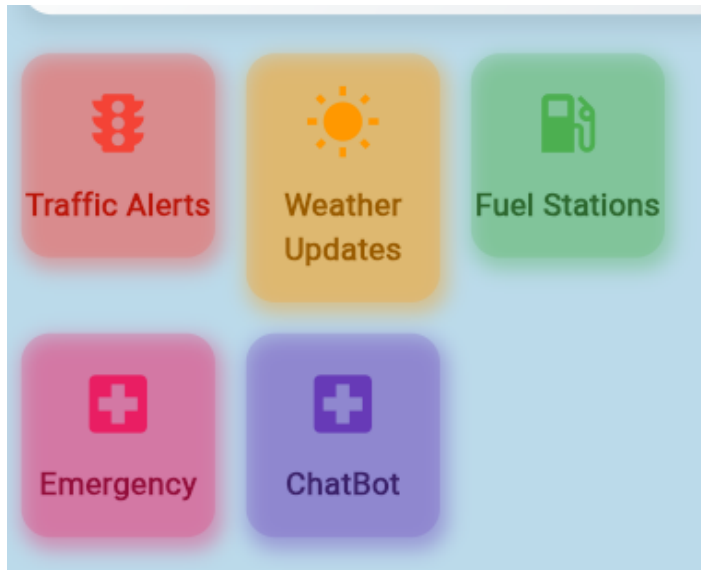
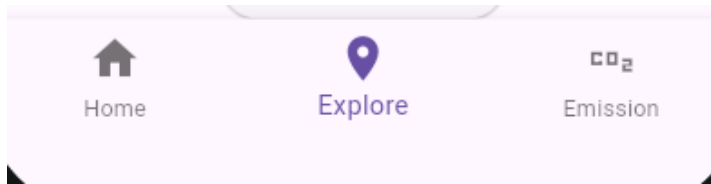
```
- Text(  
  "Hello, Flutter!",  
  style: TextStyle(fontFamily: "CustomFont", fontSize: 20, fontWeight: FontWeight.bold),  
)
```

main.dart file

```
final markers = <Marker>[  
  // Current Location Marker  
  Marker(  
    width: 50.0,  
    height: 50.0,  
    point: currentLocation!,  
    child: const Icon(Icons.location_pin, color:  
Colors.blue, size: 40),  
  ),  
  
  // Searched Location Marker  
  if (searchedLocation != null)  
    Marker(  
      width: 50.0,  
      height: 50.0,  
      point: searchedLocation!,  
      child: const Icon(Icons.location_pin, color:  
Colors.red, size: 40),  
    ),  
  
  // Turn Markers  
  ...turnMarkers.map(  
    (turn) => Marker(  
      width: 30.0,  
      height: 30.0,  
      point: turn["location"],  
      child: _getTurnIcon(turn["turnType"]),  
    ),  
  ),  
];
```

```
Icon _getTurnIcon(String turnType) {  
  switch (turnType.toLowerCase()) {  
    case "turn-right":  
      return Icon(Icons.arrow_right_alt, color:  
Colors.green, size: 30);  
  
    case "turn-left":  
      return Icon(Icons.arrow_left, color:  
Colors.blue, size: 30);  
  
    case "u-turn":  
      return Icon(Icons.u_turn_left, color:  
Colors.orange, size: 30);  
  
    case "straight":  
      return Icon(Icons.arrow_upward, color:  
Colors.grey, size: 30);  
  
    case "roundabout":  
      return Icon(Icons.sync, color:  
Colors.purple, size: 30);  
  
    default:  
      return Icon(Icons.circle, color:  
Colors.black, size: 20);  
  }  
}
```

Output :-



### Conclusion : -

Flutter offers powerful and flexible support for incorporating visual elements like icons and images to enhance UI design. Developers can easily use **built-in Material Icons** for common UI elements, or integrate **custom icons** using packages like `font_awesome_flutter` or `flutter_svg` for more design flexibility. Additionally, images in Flutter can be loaded from various sources such as assets, network, or memory, allowing dynamic and visually appealing user experiences. Mastery of these tools enables developers to create rich, intuitive, and engaging mobile interfaces with ease.